TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

JESSE LAUKKANEN

JOUKKOISTAMISEEN POHJAUTUVA WEB-PAIKALLISHAKU-KONE

Diplomityö

Tarkastaja: prof. Tommi Mikkonen
Tarkastaja ja aihe hyväksytty
Tieto- ja sähkötekniikan tiedekunta-neuvoston kokouksessa 6. touko-kuuta 2015

TIIVISTELMÄ

JESSE LAUKKANEN: Joukkoistamiseen pohjautuva Web-paikallishakukone
Tampereen teknillinen yliopisto
Diplomityö, 50 sivua, 2 liitesivua
Kesäkuu 2015
Tietotekniikan diplomi-insinöörin tutkinto-ohjelma
Pääaine: Ohjelmistotuotanto
Tarkastaja: professori Tommi Mikkonen

Avainsanat: paikallishakukone, Web, joukkoistaminen, verkkoyhteisö

Web sisältää valtavan määrän tietoa, jolla on usein maantieteellinen luonne. Samaan aikaan käyttäjät myös etsivät Webistä yhä useammin tietoa omasta ympäristöstään. Vaikka Web-dokumentit sisältävät suoria viittauksia maantieteellisiin sijainteihin, eivät nykyiset paikallishakukoneet pysty mallintamaan Webin maantieteellistä luonnetta tehokkaasti. Tästä johtuen Webin monipuolisen tiedon etsiminen maantieteellisen sijainnin perusteella ei ole mahdollista.

Tässä diplomityössä kuvataan esimerkkitoteutus joukkoistamiseen pohjautuvasta paikallishakukoneesta. Joukkoistamalla on mahdollista kerätä Web-dokumentteihin liitettävää paikkatietoa suuressa mittakaavassa. Käyttäjiltä kerätty paikkatieto myös kuvastaa ennen kaikkea juuri käyttäjien näkemystä Web-dokumenttien ja maantieteellisten sijaintien yhteydestä.

Joukkoistamisen haasteena on kuitenkin verkkoyhteisön luominen ja käyttäjien tuottaman paikkatiedon laatu. Kuten hyvin tiedetään, kaikki käyttäjät eivät aina tuota laadukasta sisältöä ja joukkoon mahtuu myös käyttäjiä, jotka pyrkivät tietoisesti käyttämään järjestelmää hyväkseen.

Työssä kuvatun esimerkkitoteutuksen pohjalta on toteutettu yksinkertainen prototyyppi, jonka avulla on kokeiltu käytännössä joukkoistamiseen pohjautuvan paikallishakukoneen toimivuutta. Prototyyppi ei sisällä kaikkia esimerkkitoteutuksessa kuvattuja ominaisuuksia. Sen perusteella voidaan kuitenkin todeta, että joukkoistaminen voi tarjota ratkaisun yleiskäyttöiseen Web-paikallishakukoneeseen, mutta ratkaisun toimivuuden toteaminen vaatii vielä lisätutkimusta.

ABSTRACT

JESSE LAUKKANEN: Crowdsourcing based local Web search engine
Tampere University of Technology
Master of Science Thesis, 50 pages, 2 Appendix pages
June 2015
Master's Degree Programme in Information Technology
Major: Software Engineering
Examiner: Professor Tommi Mikkonen

Keywords: local search, Web, crowdsourcing, online community

The information in Web often includes references to geographical locations. At the same time, users are also searching for information about their surrounding from the Web more often than before. Although Web has this huge amount of geospatial information in it, current local search engines are not modeling the geospatial nature of the Web very effectively. Because of that, it is not possible to find diverse kind of information from the Web with a spatially oriented query.

In this thesis, a reference design of a crowdsourced local search engine is described. Crowdsourcing provides a scalable way to gather spatial information related to Web-documents. Crowdsourced spatial information also represents specifically the crowd's view of the relationship between Web-documents and geographical locations.

Creating and fostering the online community and managing the quality of crowdsourced data, is the biggest challenge in crowdsourcing. As is well known, user generated content is not always top quality and there is always some members of the community that try to exploit the system for selfish reasons.

A prototype of the reference design has been implemented and used to test the possibilities of crowdsourced local search engine. The prototype does not have all the features described in the reference design. Nevertheless, experience acquired from the prototype shows that crowdsourcing might offer an effective solution to implement a general purpose Web local search engine, but thorough evaluation of the idea needs more research.

ALKUSANAT

Tämän työn tekeminen on ollut pitkä ja monivaiheinen, kärsivällisyyttä, sinnikkyyttä ja joustamista vaatinut oppimisprosessi. Haluan kiittää kaikesta ajasta ja tuesta, jonka vaimoni Tiina Laukkanen ja esikoisemme Fiona Laukkanen ovat minulle antaneet. Kiitän erityisesti myös professori Tommi Mikkosta, joka kannusti liikkeelle työn teossa ja oli valmis joustamaan merkittävästi työn aikataulutuksessa.

Kiitos kuuluu myös yliopistojärjestelmälle, joka kannustimineen sai minut lopulta saattamaan tämän työn loppuun. Näin jälkikäteen voin todeta, että työ, joka ennalta käsin näytti merkityksettömältä, on opettanut merkityksellisiä asioita kirjoittamisesta, omasta koulutusalastani, mielenkiintoisesta aiheesta ja ennen kaikkea asioiden loppuun saattamisen ilosta.

Turussa, 18.5.2015

Jesse Laukkanen

SISÄLLYSLUETTELO

KUVALUETTELO

1. JOHDANTO

Nykyaikaiset Web-hakukoneet perustuvat pääsääntöisesti avainsanoihin. Avainsanahaun avulla käyttäjä etsii tiettyihin sanoihin liittyvää tietoa World Wide Webistä (Web). Avainsanojen perusteella hakukone tuottaa listan hakutuloksia tietyssä paremmuusjärjestyksessä. Hakutulosten paremmuusjärjestys perustuu erilaisiin usein salaisiin järjestysalgoritmeihin. Googlen menestys hakukonemarkkinoilla perustuu pitkälti sen antamien hakutulosten laatuun. Googlen antamat hakutulokset vastaavat sisällöltään erittäin hyvin annettuja avainsanoja ja käyttäjän toivomia tuloksia.

Mobiililaitteilla tehtyjen hakujen yleistyessä ihmiset etsivät Webistä yhä useammin omaan ympäristöönsä liittyvää tietoa. Käyttäjä haluaa esimerkiksi etsiä tietoa lähellä olevista ravintoloista tai läheisen nähtävyyden historiasta. Tällaisessa tapauksessa hakukoneen hakutuloksia tulisi priorisoida hakutulosten maantieteellisen sijainnin perusteella, mikä edellyttää, että haun ja Web-dokumenttien maantieteellinen luonne ja sijainti ymmärretään. Hakukoneen täytyy pystyä liittämään Web-dokumentteihin paikkatietoa. Haasteena on kuitenkin tällaisen Web-dokumentteihin liittyvän paikkatiedon kerryttäminen.

Suuri osa Webin valtavasta tiedon määrästä liittyy jollakin tapaa johonkin maantieteelliseen sijaintiin. Web-dokumentit myös sisältävät suoria viittauksia maantieteellisiin sijainteihin. Esimerkiksi uutisartikkelissa kuvattuun tapahtumaan liittyy tapahtumapaikka tai blogikirjoituksessa viitataan tiettyyn paikkaan. Toistaiseksi yhteyttä Web-dokumenttien ja maantieteellisten sijaintien välillä ei kuitenkaan ole pystytty tehokkaasti ja laajamittaisesti mallintamaan (Ahlers 2012). Webin paikkatietopohjainen mallintaminen mahdollistaa niin sanotun paikallishakukoneen toteuttamisen. Paikallishakukoneen avulla käyttäjät voivat etsiä omaan ympäristöönsä liittyvää tietoa Webistä.

Nykyiset paikallishakukoneet tukeutuvat pitkälti paikkatieto- ja yritysrekistereihin, käyttäjien tuottamaan sijaintitietoon ja vain osittain Webistä eristettyyn paikkatietoon. Tästä syystä niiden avulla ei pysty etsimään monipuolisesti Webin sisältämää tietoa. Kaupallisten hakukoneiden toteutusyksityiskohdat ovat usein tarkoin varjeltuja liikesalaisuuksia ja ymmärrys niiden toiminnasta perustuu lähinnä valistuneisiin päätelmiin. (Ahlers 2012) Yksikään kaupallinen paikallishakukone ei kuitenkaan tue yleisellä tasolla tiedon etsimistä Webistä.

Tässä diplomityössä esitellään ratkaisu yleiskäyttöisen Web-paikallishakukoneen ongelmaan. Ratkaisu on yhteisöllinen joukkoistamiseen pohjautuva paikallishakukone. Webin

maantieteellinen luonne mallinnetaan verkkoyhteisön avulla. Työssä esitellään esimerkkitoteutus hakukoneesta, joka antaa verkkoyhteisölle mahdollisuuden luoda ja ylläpitää Web-dokumentteihin liittyvää paikkatietoa. Kerätyn paikkatiedon avulla käyttäjille tarjotaan mahdollisuus etsiä omaan ympäristöönsä liittyvää tietoa Webistä. Työn tavoitteena on arvioida joukkoistamiseen pohjautuvan Web-paikallishakukoneen toimivuutta.

Luvussa 2 perehdytään verkkoyhteisön luontiin, ohjaamiseen ja työn teettämiseen verkkoyhteisöllä eli joukkoistamiseen. Luvussa 3 käydään läpi hakukoneiden kehitystä ja perehdytään paikkapohjaiseen tiedon hakuun. Luvussa 4 esitellään esimerkkitoteutus yhteisöllisestä joukkoistamiseen pohjautuvasta paikallishakukoneesta. Luvussa 5 esittelemme työn ohella toteutetun paikallishakukoneprototyypin tekniset toteutusyksityiskohdat. Luvussa 6 arvioimme tässä työssä esitellyn paikallishakukoneratkaisun toimivuutta.

2. VERKKOYHTEISÖT

Webin käyttäjät ovat tiedon kuluttajia. Käyttäjät etsivät ja kuluttavat tietoa liittyen työhön, harrastuksiin, säähän, terveyteen, karttoihin ja mitä moninaisimpiin tarkoituksiin. Toisin kuin Webin alkuaikoina, tänä päivänä sosiaalisen median, foorumien ja blogien myötä iso osa Webin käyttäjistä myös osallistuu tiedon luontiin. Ihmiset keskustelevat, opiskelevat, tekevät töitä, jakavat kuvia ja videoita, pelaavat pelejä yhdessä ja etsivät uusia tuttavuuksia Webin avulla. Sosiaalisen kanssakäymisen myötä verkkopalveluiden ympärille muodostuu käyttäjien muodostamia verkkoyhteisöjä.

Verkkoyhteisöistä puhuttaessa on syytä tuntea käsite sosiaalinen media. Sosiaalisella medialla tarkoitetaan sivustojen joukkoa, jotka tarjoavat tilan syvälliselle sosiaaliselle vuorovaikutukselle, yhteisön muodostamiselle sekä yhteistyöprojekteille. Verkkoyhteisöt ovat siis osa sosiaalista mediaa. Usein näihin käsitteisiin liitetään myös termi Web 2.0, jolla viitataan sosiaalisen median teknologiseen puoleen ja varsinaisiin teknologioihin, jotka mahdollistavat interaktiiviset Web-sivustot. (Bruns & Bahnisch 2009: s. 8 - 10.)

Verkkoyhteisöille yhteistä on käyttäjien osallistuminen sisällöntuotantoon. Käyttäjien luomaan sisältöön perustuvan verkkopalvelun menestyminen on täysin riippuvainen verkkoyhteisön toiminnasta. Tästä syystä verkkoyhteisön luominen ja ohjaaminen on osa menestyksekkään verkkopalvelun kehittämistä. Käyttäjiä voidaan kannustaa osallistumaan aktiivisesti yhteisön toimintaan ja palkita hyvästä käytöksestä. (Bruns & Bahnisch 2009.)

2.1 Verkkoyhteisön luominen

Verkkoyhteisöt kehittyvät vaiheittain ja kussakin vaiheessa yhteisöllä on yksilöllisiä piirteitä ja tarpeita. Verkkoyhteisöjä luodessa yksilöiden ja koko yhteisön tarpeet tulee ottaa huomioon kussakin yhteisön eri kehitysvaiheessa. Verkkoyhteisön elinkaari voidaan jakaa esimerkiksi viiteen vaiheeseen, jotka ovat alku, luominen, kasvu, kypsyys ja kuolema (Iriberri & Leroy 2009).

Iriberrin ja Leroyn (2009) mukaan yhteisön ensimmäinen vaihe alku kuvastaa idean syntyä. Idea verkkoyhteisöstä syntyy ihmisten tarpeesta jakaa tai tuottaa yhdessä merkityksellistä tietoa. Tarpeen ja kyvyn yhdistyessä syntyy visio verkkoyhteisöstä. Vision lisäksi alkavalla yhteisöllä on käyttäytymis- ja kommunikointisääntöjä joilla varmistetaan yhteisön fokus. Käyttäytymis- ja kommunikointisäännöillä määritellään yhteisön yhteinen suunta. Fokus kuvastaa yhteisön perimmäistä tarkoitusta ja päämäärää. Fokus voi olla esimerkiksi uuden teknologian seuraaminen, jolloin yhteisön pelisäännöt, olivatpa ne kirjoitettuja tai kirjoittamattomia, määrittelevät, että keskustelun on syytä liittyä läheisesti

juuri uuden teknologian seuraamiseen. Usein säännöt kirjoitetaan selkeästi kaikkien saataville, mutta toisinaan säännöt ilmenevät ainoastaan yhteisön jäsenten käyttäytymisestä.

Vision ja fokuksen synnyttyä yhteisön seuraava luonteva askel on ottaa käyttöön tarvittavaa teknologiaa. Teknologia valinnat tehdään ensimmäisten jäsenten ja potentiaalisten jäsenten tarpeiden perusteella. Teknologian mahdollistaessa yhteisön jäsenten välisen kommunikoinnin ja toiminnan, yhteisön voidaan katsoa siirtyneen luomisvaiheeseen (Iriberri & Leroy 2009). Teknologioiden hyödyntäminen voi tarkoittaa yksinkertaisen foorumin käyttöönottoa tai kokonaisten sivustojen rakentamista. Keskeistä luomisvaiheessa on yhteisön siirtyminen visiosta käytännössä toimivaksi yhteisöksi. Nimitys verkkoyhteisö tulee nimenomaan verkkoteknologian hyödyntämisestä.

Brunsin ja Bahnischin mukaan ensimmäisten käyttäjien toimintaa on tärkeää ohjata luomalla esimerkki sisältöä ja antamalla esimerkkejä toivotun kaltaisista toiminnoista. Näin asetetaan ensinnäkin sisällön laadulle lähtökohta ja toisekseen annetaan esimerkkejä hyvistä toimintamalleista. Tämän lisäksi Bruns ja Bahnisch korostaa, että ensimmäisten käyttäjien joukko olisi syytä olla rajattu esim. suljetun beta-testauksen muodossa tai muuten kontrolloitu joukko, jotta voidaan varmistaa käyttäjien toiminnan olevan halutun kaltaista. (Bruns & Bahnisch 2009 s. 19) Uudet käyttäjät kopioivat aikaisempien käyttäjien toimia ja tästä syystä ensimmäisten käyttäjien toimet ovat erityisen tärkeässä asemassa. Mitä kontrolloidumpi yhteisön liikkeellelähtö on, sitä paremmin saadaan varmistettua, että tuleville käyttäjämassoille asetettu esimerkki on halutun kaltainen.

Yhteisön jäsenmäärän kasvaessa yhteisön identiteetti ja kulttuuri alkaa muodostua. Tähän vaiheeseen kuuluu, että jäsenet alkavat käyttää yhteisiä käsitteitä, valita roolejaan yhteisössä sekä osallistumiseen liittyvä etiketti alkaa muodostua entistä vahvemmin. Verkkoyhteisön kasvuvaiheet vastaavat hyvin tarkasti fyysisten yhteisöjen kasvua. Osa jäsenistä ottaa johtavan roolin luoden sisältöä ja johtaen keskusteluja, kun taas suuri osa jäsenistä ottaa passiivisen roolin pääasiassa kuluttaen tietoa. (Iriberri & Leroy 2009) Lähes kaikille verkkoyhteisöille yhteistä on, että valtaosa yhteisön jäsenistä ei osallistu aktiivisesti sisällöntuotantoon ja valtaosan sisällöstä tuottaa pieni käyttäjien vähemmistö. Sivuston käytettävyydellä ei ole merkittävää vaikutusta siihen osallistuuko käyttäjät sisällöntuotantoon (Cliff et al. 2010).

Yhteisön kasvaessa käyttäjien luoman sisällön määrä kasvaa myös. Tämä aiheuttaa yhä kovempia paineita sisällön moderoinnille. Moderoinnilla tarkoitetaan ei toivotun sisällön ja toimintatapojen karsimista. Bruns ja Bahnich korostaa moderoinnin haasteellisuutta. Yhtenä keskeisimpänä ajatuksena haasteen ratkaisemiseksi he mainitsevat uusien käyttäjien yhteisöllistämisen. Yhteisöllistämisellä tarkoitetaan yhteisön etiketin ja sääntöjen opettamista uusille käyttäjille siten, että yhteisön fokus pysyy eikä uudet käyttäjät ala luomaan omia sääntöjään jo olemassa olevien tilalle. (Bruns ja Bahnich 2009 s. 20–21.)

Jos ajatellaan, että moderoinnin päätehtävä on säilyttää yhteisön fokus, niin yhteisön fokuksen korostaminen uusille käyttäjille voidaan nähdä ennaltaehkäisevänä toimintana. Moderoinnissa tärkeintä on ennaltaehkäistä ongelmia.

Yhteisön kypsyessä ja koon kasvaessa tarvitaan yhä muodollisempaa ja selkeämpää sääntelyä ja käyttäjien palkitsemista fokuksen säilyttämiseksi. Yhteisön kypsymisvaiheessa vanhat kävijät menettävät kiinnostuksensa yhteisöön ja uusia jäseniä liittyy. Yhteisön jäsenten vaihtuessa myös yhteisön tarpeet muuttuvat. Verkkopalvelun täytyy muuttua jäsenten mukana ja tarjota mielenkiintoisia ja palkitsevia tapoja osallistua yhteisöön aktiivisina jäseninä. Parhaimmillaan yhteisö voi pysyä kypsyysvaiheessa pitkään, mutta lopulta jäsenten kyllästyessä yhteisöön aktiivinen osallistuminen vähenee ja sisällön laatu heikkenee, joka lopulta johtaa yhteisön kuolemiseen. (Iriberri & Leroy 2009.)

2.2 Verkkoyhteisön ohjaaminen

Verkkoyhteisö on täysin riippuvainen käyttäjien aktiivisuudesta. Ilman aktiivisia käyttäjiä yhteisö kuolee. Aktiiviset käyttäjät tuottavat yhteisölle arvoa, joka taas houkuttelee uusia käyttäjiä yhteisöön. Ludford et al. (2004) toteaa, että aktiivisuus yhteisössä luo lisää aktiivisuutta. Ongelma onkin, miten luoda aktiivisuutta yhteisön alkutaipaleella. Yhteisön kasvaessa aktiivisten käyttäjien kokonaismäärä kasvaa ja sisältöä on entistä paremmin saatavilla. Näin toimii terve verkkoyhteisö. Verkkoyhteisön yhtenä tärkeimpänä tavoitteena voidaan pitää siis aktiivista käyttäjäpohjaa.

Aktiivinen käyttäjäpohja ei yksistään riitä verkkoyhteisön menestymiseen, vaan osallistumisen täytyy olla myös verkkoyhteisön tavoitteiden mukaista ja laadukasta. Esimerkiksi aktiivinen spämmiviestien kirjoittaja ei vie yhteisöä eteenpäin vaan voi saada jopa tuhoisia vaikutuksia aikaiseksi yhteisössä. Osallistumisen laatu kuvaa siis, kuinka hyvin käyttäjien toimet edistävät yhteisön yhteisiä tavoitteita. Jotta osallistumisen laatua voidaan mitata, täytyy yhteisön tavoitteet olla selkeästi määriteltyjä.

2.2.1 Käyttäjien aktiivisuus

Lampe et al. (2010) mukaan valtaosa verkkoyhteisöjen jäsenistä ei osallistu sisällön tuottamiseen vaan ottavat passiivisen roolin keskittyen vain tiedon hyödyntämiseen - ei sen tuottamiseen. Passiivisen roolin omaksuvia käyttäjiä pidetään yhteisön vähemmän arvostettuina jäseninä ja heitä kutsutaan usein vapaamatkustajiksi. Vapaamatkustajat nähdään usein henkilöinä jotka hyötyvät muiden aktiivisuudesta antamatta mitään takaisin. Ongelma tunnetaan myös nimellä yhteismaan ongelma (engl. tragedy of the commons). Tämän ongelman ratkaisemiseksi yhteisöä pitää valvoa ja huonosta käyttäytymisestä täytyy koitua seurauksia. Menestyville yhteisöille on yhteistä, että yhteisön jäsenet huolehtivat itse yhteisön valvomisesta ja sanktioista. Verkkoyhteisöissä yhteisön seuraaminen on helpompaa verrattuna muihin yhteisöihin, mutta toisaalta sanktioiden käyttäminen on entistä haastavampaa. (Kollock & Smith 1996.)

Ludford et al. toteavat tutkimuksessaan, että verkkoyhteisön jäseniä voidaan inspiroida osallistumaan aktiivisemmin sosiaalipsykologiasta tutuilla keinoilla. Yhteisön jäsenet pitävät muun muassa, kun heille kerrotaan kuinka yksilöllisiä he ovat yhteisössä ja kuinka heidän osallistumisensa hyödyttää yhteisöä ainutlaatuisella tavalla. Yhteisön jäsenten yksilöllisyyden korostaminen lisää jäsenten aktiivisuuttaan yhteisössä. (Ludford et al. 2004)

Positiivisen palautteen antaminen ja jäsenten huomioiminen yksilöllisellä tavalla tuntuu luontevalta tavalta palkita ja kannustaa jäseniä toimissaan. Käyttäjille voidaan esimerkiksi näyttää yksilöllistä tietoa heidän tuottamastaan sisällöstä ja tietoa siitä kuinka moni muu käyttäjä on kuluttanut tai hyötynyt heidän tuottamastaan sisällöstä. Käyttäjien motiivi osallistua yhteisön toimintaan voi kuitenkin muuttua ajan myötä. Lampe et al. toteaa, että yksi keskeisimpiä motivaatiotekijöitä käyttäjien osallistumiselle on käyttäjän tunne kuulumisesta yhteisöön. Sosiaalisilla ja kognitiivisilla tekijöillä oli myös merkittävämpi vaikutus osallistumiseen kuin käytettävyydellä. (Lampe et al. 2010)

Nonnecke et al. (2006) kutsuvat tutkimuksessaan passiivisten käyttäjien joukkoa luurailijoiksi (lurker). Tutkimuksessaan he toteavat, että luurailijoiden tyytyväisyys verkkoyhteisöä kohtaan oli selkeästi huonompi kuin aktiivisesti sisällön luontiin osallistuneilla käyttäjillä. Epäselväksi kuitenkin jäi, johtuiko luurailijoiden tyytymättömyys passiivisesta käyttäytymisestä vai johtuiko tyytymättömyys yhteisön käytöksestä, joka johti passiivisuuteen. Nonnecke et al. kannustaa yhteisön ylläpitäjiä aina selvittämään syitä luurailukäyttäytymisen takana. Kun passiivisen käyttäytymisen perusteet ovat tiedossa, osataan paremmin tukea niitä käyttäjiä, jotka haluaisivat osallistua yhteisöön, mutta eivät teknisestä tai sosiaalisista syistä kuitenkaan osallistu. Toisaalta passiivinen käyttäytyminen on täysin hyväksyttävä osa yhteisöä jolloin myös passiivisille käyttäjille tulee taata mahdollisuus kokea kuuluvansa joukkoon.

2.2.2 Käyttäjien tunnistaminen

Verkkoyhteisössä yksittäisiin käyttäjiin viitataan pääsääntöisesti joko nimimerkillä eli käyttäjän pseudonyymillä tai käyttäjän nimellä. Esimerkiksi Facebook vaatii rekisteröinnin yhteydessä käyttäjältä oikean nimen antamista ja kieltää käyttöehdoissa kaiken virheellisen tiedon antamisen (Facebook 2015). Webissä historiassa perinteinen tapa on ollut käyttää nimimerkkiä. Webin anonymiteetin purkautuessa usealla tasolla myös oikean nimen käyttämisestä on tullut yleisempää. Teknisesti ajatellen oikean nimen antamista on vaikea pakottaa ja käyttäjän näkökulmasta ainoa tekninen ero on lähinnä rekisteröitymisvaiheessa täytetyssä käyttäjätietolomakkeessa, jossa joko kysytään käyttäjän etu- ja sukunimeä tai nimimerkkiä.

Nimimerkki antaa käyttäjälle osittaisen anonymiteetin, koska lähtökohtaisesti nimimerkkiä ei voida liittää käyttäjän oikeaan identiteettiin. Kuinka vahva anonymiteetti on, riippuu monista tekijöistä. Kuten hyvin tiedetään, Webin käyttäjien toimia seurataan nykypäivänä usean tahon toimesta ja vahvan anonymiteetin saavuttaminen on yhä vaikeampaa

ja vaatii yhä enemmän teknistä osaamista. Nimimerkin käyttäminen tuo kuitenkin käyttäjille anonymiteetin tunteen, joka vaikuttaa käyttäjän toimiin (Cho & Kim 2012). Oikean nimen käyttäminen on selkeämmin yhdistettävissä käyttäjän oikeaan identiteettiin eikä näin ollen anna käyttäjälle anonymiteetin suojaa.

O'Keefe (2011) listaa kirjoituksessaan seuraavia kokemuksia nimimerkkikäytännön vaikutuksista:

- Käyttäjälle jää mahdollisuus valita oma identiteettinsä yhteisössä.
- Käyttäjä voi halutessaan valita oikean nimensä nimimerkiksi.
- Nimimerkki tarjoaa osittaisen anonymiteetin.
- Anonymiteetti tuo mukanaan vastuuttomuutta.
- Anonyymi käyttäjä kokee olevansa vähemmän vastuussa tekemisistään ja näin ollen anonyymiys lisää käyttäjien asiatonta käyttäytymistä.
- Anonymiteetti ei sovellu ammattimaiselle yhteisölle.

Oikean nimen käyttämisestä O'Keefe listaa seuraavia huomioita:

- Omaa nimeä käytettäessä ihmiset ovat huomattavasti tarkempia tekemisistään.
- Antaa luotettavamman mielikuvan yhteisöstä.
- Miltä esimerkiksi LinkedIn vaikuttaisi, jos käyttäjät tunnettaisiin vain nimimerkillä?
- Käyttäjät eivät voi hyödyntää jo luotua Internet identiteettiään.
- Yhteisössä saattaa olla useita samannimisiä jäseniä.
- Anonymiteetin puuttuminen lisää kynnystä osallistua joillakin käyttäjillä.

Toisaalta Cho & Kim (2012, ss. 3046–3047) eivät nähneet tutkimuksessaan eroa osallistumisaktiivisuudessa oikean nimen käyttämisen ja nimimerkkikäytännön välillä. Anonymiteetin tunteella on varmasti merkitystä kuinka herkästi yksilö uskaltaa ilmaista itseään, mutta toisaalta oikean nimen käyttäminen voi kenties myös kannustaa aktiivisuuteen. Onhan oikealla nimellä ansaittu kunnia suoremmin yhteydessä muuhun elämään.

Oikeiden nimien käyttämisellä käyttäjänimien sijaan on merkittävä vaikutus käyttäjien käyttäytymiseen. Oikean nimen käyttäminen vähentää merkittävällä tavalla käyttäjien estotonta käyttäytymistä (Cho & Kim 2012, s. 3047). Estoton käyttäytyminen johtaa epäasiallisen sisällön tuottamiseen.

O'Keefen kokemuksien ja Cho & Kimin tutkimustulosten pohjalta voidaan todeta, että oikean nimen käyttäminen johtaa asiallisempaan ja vastuullisempaan käyttäytymiseen verkkoyhteisössä. Perustettaessa verkkoyhteisöä, jossa käyttäjien toivotaan tuottavan asiallista muuta yhteisöä hyödyttävää sisältöä, voidaan todeta, että oikean nimen käyttäminen on nimimerkkiä parempi valinta. Vaikka oikean nimen käyttämisellä olisikin vaikutusta käyttäjien aktiivisuuteen, niin sen vaikutus käyttäjien asialliseen muita kunnioittavaan käyttäytymiseen on huomattavasti suurempi.

2.3 Joukkoistaminen

Joukkoistamiselle (engl. crowdsourcing) on kirjallisuudessa useita eri määritelmiä[1], eikä yhtä vakiintunutta määritelmää ole (Hosseini et al. 2014). Tässä työssä joukkoistamisella tarkoitetaan toimintamallia, jossa työ puretaan tehtäviksi ja teetetään suurella väkijoukolla. Tämä määritelmä on hyvin lähellä Baba & Kashiman (2013) määritelmää, mutta poikkeaa siinä, että Baba & Kashima mieltää joukkoistamisen vain verkossa tapahtuvaksi toiminnaksi.

Joikkoistamisessa on kyse työn ulkoistamisesta ihmismassoille. Ihmisiä voidaan kannustaa osallistumaan esimerkiksi palkitsemalla osallistumisesta suoraan tai välillisesti. Osallistujat voivat tehdä työtä yksin tai yhteistyössä muiden kanssa. Joukkoistamisen on katsottu soveltuvan erityisesti tehtäviin, jotka ovat helposti yleisön ymmärrettävissä, mutta samalla vaativat useita eri näkökulmia ja suuren joukon ongelmanratkaisijoita. (Hosseini et al. 2014.)

Joukkoistamisen neljä peruskäsitettä joukko (engl. crowd) eli työnsuorittajat, joukkoistaja (engl. crowdsourcer) eli työn teettäjä, joukkoistettu työ (engl. crowdsourced task) ja joukkoistamisalusta (engl. crowdsourcing platform) eli järjestelmä, joka mahdollistaa joukkoistamisen, kuvastavat tieteenalan laajuutta (Hosseini et al. 2014). Kyse ei ole ainoastaan tietojärjestelmistä tai työn teettämisestä vaan kyseessä on monipuolisesti teknologiaa ja ihmismassoja yhdistävästä mallista. Tästä syystä joukkoistamiseen läheisesti liittyviä aiheita ovat: tietojenkäsittelytiede, ihmisen ja tietokoneen vuorovaikutus, työ ja organisaatio ekonomia, psykologia, ja sosiologia, kuten myös poliittiset, lainsäädännölliset ja eettiset näkökulmat (Chi & Bernstein 2012).

Koska joukkoistaminen on niin laaja ja monisyinen aihealue, tässä työssä ei voida paneutua syvällisesti joukkoistamisen teoriaan. Sen sijaan joukkoistamisesta esitellään keskeisimmät piirteet, mahdollisuudet ja haasteet.

2.3.1 Käyttäjien luoma sisältö

Ihmiset ovat aina olleet kriittinen osa tietojärjestelmiä, mutta yhä useampi järjestelmä tukeutuu käyttäjämassojen kykyyn tuottaa sisältöä. Ihmisjoukot luovat yhdessä tietosanakirjoja, kuten Wikipedia (https://www.wikipedia.org/), karttoja, esimerkiksi OpenStreetMap (https://www.openstreetmap.org), vastaavat toistensa kysymyksiin, kuten Stack Overflow:ssa (http://stackoverflow.com/), arvioivat tuotettua sisältöä, suunnittelevat t-paitoja, kuten Threadless-palvelussa (https://www.threadless.com/), auttavat tutkijoita muodostamaan proteiineja, löytämään kosmista pölyä ja tuottavat käytännössä mitä tahansa sisältöä (Lukyanenko 2012). Kaikki nämä järjestelmät kuitenkin painivat myös yh-

[1] Hosseini et al. viittaa työssään lisämateriaaliin, jossa on listattu eri määritelmiä joukkoistamiselle [WWW]. [Viitattu 26.4.2015]. Saatavissa: http://goo.gl/gsV5XB

teisten ongelmien kanssa. Osa käyttäjistä on rehellisiä ja vilpittömiä, mutta osa epärehellisiä ja vilpillisiä, jotka pyrkivät hyväksikäyttämään järjestelmää omien etujensa tavoitteluun.

Luotettavuus, ihmisen kyky sitoutua täyttämään muiden oikeutetusti asettamat odotukset, on perustava hyve sekä avainasemassa oleva edellytys, minkä tahansa yhteiskunnan olemassaololle (Dunn 1984). Dunn viittaa John Locken ajatuksiin luottamuksen keskeisyydestä yhteiskunnassa.

Verkkoyhteisöt ovat aivan yhtä todellisia kuin yhteisöt, joiden jäsenet tapaavat fyysisesti. Luottamus on keskeinen vakautta luova tekijä missä tahansa yhteisössä, mukaan lukien verkkoyhteisöissä. Luottamuksella on vastaava rooli verkkoyhteisössä kuin sillä on eivirtuaalisesti tapaavissa yhteisöissä. Siksi verkkoyhteisön kannalta on elintärkeää, että yhteisössä luottamusta mallinnetaan riittävällä tasolla. (Abdul-Rahman & Hailes 2000)

Mallintamalla käyttäjien mainetta, voidaan arvioida käyttäjien luotettavuutta. Käyttäjien mainetta arvioidaan ja hallitaan mainejärjestelmän avulla. Mainejärjestelmät perustuvat pääasiassa käyttäjien väliseen yhteistyöhön, tiedon jakamiseen ja keskinäiseen arviointiin. Alnemr ja Meinel (2011) listaavat kirjallisuudesta ja kokemuksistaan johtamiaan ohjeita mainejärjestelmän rakentamisesta seuraavasti:

1. Määrittele mihin maineen käsitettä tarvitaan järjestelmässä.
2. Tunnista entiteetit, jotka muodostava järjestelmän.
3. Määrittele miten järjestelmän entiteetit identifioidaan.
4. Päätä millä entiteeteillä on maine.
5. Määrittele minkä tyyppinen kunkin entiteetin maine on (esim. tasot, pisteet, arvostelut).
6. Päättele kriteerit, jotka muodostavat kunkin entiteetin maineen.
7. Selvitä mitä tietoja tarvitaan, jotta kullekin entiteetille voidaan määrittää sen maine ja miten tiedot saadaan kerättyä.
8. Valitse sopiva algoritmi maineen laskemiseen.
9. Päätä miten maine näytetään käyttöliittymässä, jos sellainen on.
10. Määrittele toiminnot maineen hallintaan.
11. Määrittele käyttöoikeudet (kuka näkee ja mitä näkee).
12. Selvitä muita maineeseen vaikuttavia tekijöitä.
13. Määrittele miten ajankulku vaikuttaa maineeseen.

Listatut askeleet antavat suuntaa mainejärjestelmän perusteiden rakentamiseen. Laadukkaan mainejärjestelmän rakentaminen vaatii edellä mainittujen lisäksi toimialakohtaista suunnittelua. (Alnemr & Meinel 2011.)

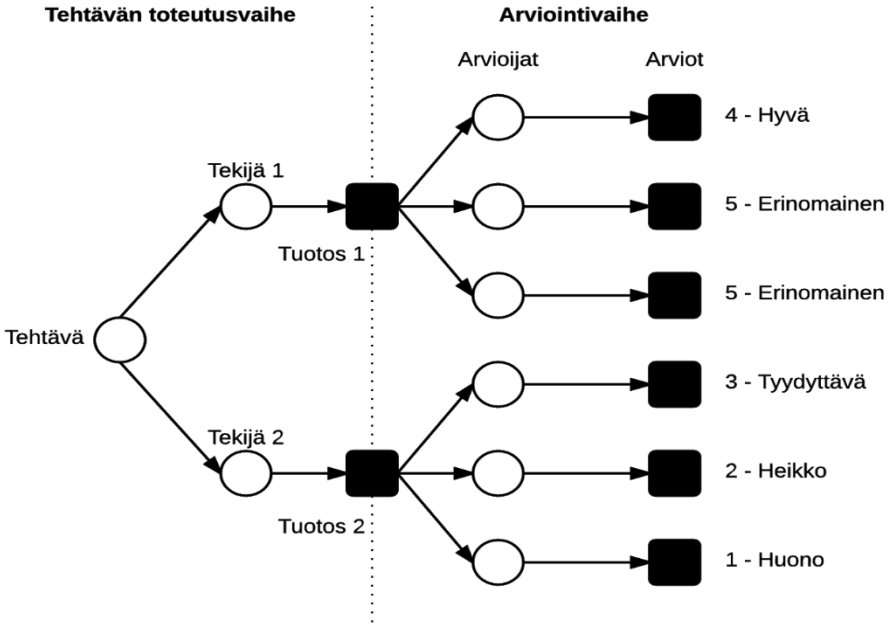
2.3.2 Laadunhallinta

Joukkoistamiseen perustuvien järjestelmien yksi suurimmista ongelmista on laadunhallinta. Käyttäjät eivät välttämättä ole kykeneviä saati motivoituneita laadukkaan sisällön tuottamiseen. Osa käyttäjistä pyrkii tietoisesti suoriutumaan tehtävistä minimaalisella

työllä mahdollisen palkinnon toivossa. (Baba & Kashima 2013) Strong el. al. (1997) tunnistaa datan tuotannossa toistuvan kolme keskeistä roolia: tuottajat, valvojat ja kuluttajat. Datan valvojilla tarkoitetaan niitä, jotka huolehtivat datan tallentamisesta, ylläpidosta ja turvaamisesta. Datan laatua tutkivien tutkijoiden keskuudessa yleisesti hyväksytty korkealaatuisen datan määritelmä on: "data, joka soveltuu datan kuluttajien käyttötarpeisiin" (Lee 2003).

Perinteisesti tietojärjestelmät ovat tukeutuneet organisaatioiden ja yritysten itse tuottamaan ja valvomaan dataan. Datan luojia pystyttiin kouluttamaan ja ohjeistamaan, heille voitiin antaa rakentavaa palautetta. Sosiaalisen median ja Web 2.0:n myötä yhä useammat järjestelmät perustuvat käyttäjien luomaan sisältöön. Käyttäjien luodessa dataa, perinteiset datan laadunvarmistusprosessit eivät enää toimi. Käyttäjämassoja ei ole käytännöllistä kouluttaa tai helppoa opastaa. Myös datan luojien ja kuluttajien välinen raja hämärtyy. Samat käyttäjät sekä luovat että kuluttavat dataa ja itse luodun datan laadun arviointi on hankalaa. Perinteiset informaation laadunhallinnan keinot eivät vain toimi ympäristössä, jossa käyttäjät vastaavat sisällön tuotannosta. (Lee 2003.)

Joukkoistamalla tuotetun datan laatua voidaan arvioida esimerkiksi redundanssin avulla eli laittamalla useampi käyttäjä tekemään samaa tehtävää, jolloin tuotettua dataa on mahdollista arvioida tilastollisesti esimerkiksi keskiarvo tai enemmistöperiaatteella tai jollain monimutkaisemmalla tavalla. Tilastollinen lähestymistapa toimii, kun dataa kerätään rakenteellisessa muodossa, esimerkiksi monivalintakysymykset tai kyllä/ei -äänestys. Lähestymistapa ei sovellu kuitenkaan monimutkaisempiin tehtäviin, kuten logosuunnitteluun tai kirjoittamiseen. (Baba & Kashima 2013, s. 554–555.)



Kuva 1. *Kaksivaiheinen työnkulku (Baba & Kashima 2013).*

Monimutkaisempien tehtävien arviointiin voidaan käyttää kuvassa 1 esitettyä kahden vaiheen menetelmää. Ensimmäisessä vaiheessa usea käyttäjä suorittaa annetun tehtävät. Toisessa vaiheessa arvioidaan ensimmäisen vaiheen tuotoksia. Arviot suoritetaan esimerkiksi monivalintakysymyksillä. Näin saadaan tilastollisesti arvioitavaa arviointidataa, jonka perusteella voidaan epäsuorasti arvioida ensimmäisen vaiheen tuotoksia. Tätä mallia voidaan vielä tehostaa painottamalla tuloksia tehtävien suorittajien kyvykkyyden ja arvioijien taipumusten mukaan, sikäli kun käyttäjistä on tällaista tietoa käytettävissä. (Baba & Kashima 2013.)

3. HAKUKONEET

Web on maailmanlaajuinen, monikielinen ja alati kasvava verkko. Tästä syystä tietoa löytyy lähes mihin tahansa tarpeeseen. Ongelmana on kuitenkin merkityksellisen tiedon löytäminen tehokkaasti. Tätä ongelmaa ratkaisemaan on kehitetty moninainen joukko hakukoneita. Hakukoneet tarjoavat käyttäjille keinon etsiä haluamaansa tietoa Webistä. Hakukoneet voidaan karkeasti jakaa yleiskäyttöisiin hakukoneisiin, esimerkiksi Google (https://www.google.com), Bing (https://www.bing.com), Yahoo (https://www.yahoo.com), DuckDuckGo (https://duckduckgo.com), ja erikoistuneisiin hakukoneisiin. Muutamana esimerkkinä erikoistuneista hakukoneista voidaan mainita esimerkiksi tieteellisiin julkaisuihin keskittyneet hakukoneet, kuten Google Scholar (https://scholar.google.com), IEEE Xplore (http://ieeexplore.ieee.org), yrityshakukoneet kuten, Yelp (http://www.yelp.com), Google Maps (https://www.google.com/maps/), Bing local (https://www.bing.com/local/), YellowPages (http://www.yellowpages.com) ja lentohakukoneet, kuten Google Flights (https://www.google.com/flights/), Momondo (http://www.momondo.com), Skyscanner (http://www.skyscanner.com). Erikoistuneita hakukoneita löytyy valtava määrä. Tässä työssä mielenkiinnon kohteena on erityisesti paikallishakukone, joka on itsessään tietyn tyyppinen erikoistunut hakukone (Ahlers 2012). Paikallishakukoneella tarkoitetaan tässä työssä erityisesti yleiskäyttöistä paikallishakukonetta.

Tässä luvussa käydään läpi hakukoneiden kehitystä ja nykyistä tilaa. Ensiksi käydään läpi yleiskäyttöisten hakukoneiden kehitys, jonka jälkeen perehdymme paikallishakukoneisiin.

3.1 Hakukoneiden kehitys

Webin varhaisessa kehitysvaiheessa Tim Berners-Lee ylläpiti listaa Web-palvelimista. Uusien palvelimien määrä kasvoi kuitenkin nopeasti niin suureksi, että Berners-Leen lista ei pysynyt muutoksen vauhdissa mukana. Berners-Leen listaa seurasi useat muut käsin ylläpidetyt hakemistot Web-sivuista. Ensimmäisenä alkeellisena Web-hakukoneena voidaan ehkä pitää Oscar Nierstraszin kirjoittamaa Perl-skriptiä, joka keräsi käsin ylläpidetyiltä listoilta linkkejä ja listasi ne yhdenmukaisessa muodossa. Nierstraszin ohjelma loi perustan W3Catalogille. (Seymour et al. 2011)

Matthew Grayn 1993 luoma World Wide Web Wonderer oli todennäköisesti ensimmäinen Web-robotti, joka ryömi läpi Webin etsien uusia sivuja ja listaten löytämiään. Wondereria seurasi Aliweb, joka Wondererista poiketen ei käyttänyt robottia sivujen etsimiseen vaan tukeutui Web-sivujen ylläpitäjiltä saatuihin ilmoituksiin indeksoitavista si-

vuista. Myös 1993 julkaistu Jump Station oli ensimmäinen hakukone, joka yhdisti nykyaikaisen hakukoneen kolme keskeistä ominaisuutta: sivujen etsimisen ryömimällä, sivujen indeksoinnin ja hakutulosten etsimisen hakusanan avulla. Rajallisista resursseista johtuen Jump Streetin tarjoama haku rajoittui kuitenkin vain Web-sivujen otsikkoihin. WebCrawler ja Lycos (1994) olivat ensimmäisten hakukoneiden joukossa, jotka toivat tullessaan haun, joka tarjosi hakutuloksia koko Web-sivun sisällön perusteella. Tästä alkoi lukuisten hakukoneiden intensiivinen kilpailu käyttäjien ja sijoittajien suosiosta, kunnes suuri osa niistä jäi IT-kuplan jalkoihin vuosien 1999 - 2001 aikana. (Seymour et al. 2011)

Googlen ylivoima hakukonemarkkinoilla alkoi näkyä vuoden 2000 tienoilla. Google tarjosi minimalistisen käyttöliittymän ja laadukkaampia hakutuloksia kuin muut PageRank-innovaationsa ansiosta (Seymour et al. 2011). Googlen dominanssi on jatkunut koko 2000-luvun. Vuonna 2015 Googlen markkinaosuus työpöytäkoneilla tehdyistä hauista on arvioitu olevan noin 65 % ja mobiilihauista noin 91 % (Netmarketshare 4.5.2015). Muita isoja hakukoneita ovat muun muassa kiinankielinen hakukone Baidu (http://www.baidu.com/), Microsoftin Bing, Yahoo!, jonka hakutulokset Bing on tuottanut vuodesta 2009 alkaen (Seymour et al. 2011) ja venäjänkielinen Yandex (https://www.yandex.com/).

Googlen dominanssista huolimatta kilpailu hakukonemarkkinoilla on aktiivista. 2000-luvulla hakukoneet ovat pyrkineet erottumaan joukosta paremmilla ominaisuuksilla ja laadukkaammilla hakutuloksilla. Esimerkiksi Ask (http://www.ask.com/) pyrkii tukemaan luonnollisella kielellä tehtyjä hakuja ja vastaamaan suoraan käyttäjän esittämään kysymykseen. Dogpile (http://www.dogpile.com/) on esimerkki metahakukoneesta, joka pyrkii tuottamaan laadukkaampia hakutuloksia yhdistämällä useiden hakukoneiden hakutuloksia. Baidu ja Yandex ovat pystyneet haastamaan Googlen omilla kielialueillaan, joskin taustalla saattaa olla enemmän poliittiset syyt kuin paremman hakukokemuksen tarjoaminen. Tietovuotojen ja laajamittaisen verkkoseurannan seurauksena käyttäjien yksityisyyttä suojaavat hakukoneet kuten DuckDuckGo ovat nostaneet huomattavasti suosiotaan (DuckDuckGo 4.5.2015). Edellä mainittujen lisäksi muun muassa Google tarjoaa mahdollisuuden etsiä kuvan tai äänen avulla.

Sosiaalisen median myötä Webiin luodaan sisältöä joka hetki valtavat määrät. Tämä jatkuva sosiaalisen median tuottama sisältö on luonut uudentyyppisiä vaatimuksia hakukoneille, hakukoneiden pitää pystyä tarjoamaan ajantasaisia hakutuloksia. Siinä missä perinteinen hakukone indeksoi Web-dokumentteja ja tarjoaa hakutuloksia tallentamiensa tietojen perusteella, reaaliaikainen hakukone pyrkii tuottamaan hakutuloksia, jotka ovat täysin ajantasaisia. Nykyiset reaaliaikaiset hakukoneet tuottavat hakutuloksia pääasiassa sosiaalisen median sisällöstä, kuten Facebookista (https://www.facebook.com/), Twitteristä (https://twitter.com/) ja Google plussasta (https://plus.google.com). (Quazilbash et al. 2012.)

3.2 Hakukoneen toiminta

Hakukoneen tärkein tehtävä on auttaa käyttäjää löytämään etsimänsä tieto - nopeasti. Suoriutuakseen tästä tehtävästä hakukoneen täytyy tietää mitä tietoa Webissä on, missä se on ja tarjota tämä tieto käyttäjälle hakutuloksina. Hakutulokset sisältävät perinteisesti otsikon, lyhyen kuvauksen ja linkin Web-dokumenttiin, jossa tietoa käyttäjän hakuun löytyy. Hakutuloksena voidaan näyttää myös suoraa vastausta esitettyyn kysymykseen, kuten esimerkiksi Google, DuckDuckGo ja Ask pyrkii tekemään mahdollisuuksien mukaan.

Tietääkseen mitä tietoa Webissä on, hakukoneen täytyy kerätä tietoa tehokkaasti ja pitää tietonsa ajan tasalla. Tähän tarkoitukseen hakukoneet käyttävät robotteja, jotka ryömivät Webissä seuraten Web-dokumenteista löytämiään linkkejä. Ryömijärobotit välittävät löytämänsä dokumentit indeksoitaviksi. Indeksointivaiheessa dokumentit analysoidaan ja niiden sisältämä keskeinen tieto tallennetaan tietokantaan. Eri hakukoneilla on eri strategioita siihen mitä tietoa ja miten tietoa indeksoidaan. Indeksoinnin tehtävä on mahdollistaa tiedon nopea löytäminen. (Brin & Page 1998; Seymour et al. 2011.)

Indeksoidun tiedon avulla käyttäjälle voidaan tarjota hakutuloksia. Perinteisesti käyttäjä ei siis tehdessään hakua etsi tietoa Webistä vaan hakukoneen indeksistä. Hakukoneet pyrkivät päivittämään indeksiään jatkuvasti, jotta se kuvaisi Webin ajantasaista sisältöä. Vielä 1990-luvun alkupuoliskolla uskottiin, että hyvien hakutulosten tarjoamiseksi riitti, että hakukoneen indeksi sisälsi mahdollisimman suuren osan koko Webistä (Brin & Page 1998). Käytäntö kuitenkin osoitti, että vaikka hakukoneet pystyivät palauttamaan suuren määrän hakutuloksia, jotka sisälsivät annetut avainsanat, pelkkä hakutulosten suuri määrä ei hyödyttänyt käyttäjää. Ensiarvoisen tärkeää oli hakutulosten laatu.

Googlen perustajat Sergey Brin ja Larry Page toteavat artikkelissaan jo 1998, että koko Webin kattava indeksi ei yksinään takaa tiedon helppoa löydettävyyttä. Ei riitä, että hakukone löytää käyttäjää kiinnostavan hakutuloksen, jos kyseinen hakutulos ei ole hakutulosten kärkipäässä. Aidosti relevantit hakutulokset voivat siis hukkua muiden niin sanottujen roskatulosten joukkoon. Mitä enemmän roskatuloksia käyttäjälle näytetään, sitä suurempi työ käyttäjällä on vielä etsiä relevantti hakutulos roskien seasta. Relevanttien hakutulosten osuus kaikista käyttäjän katsomista hakutuloksista kuvaa hyvin hakutulosten laatua. (Brin & Page 1998.)

Brin & Page (1998) pitävät relevantteina hakutuloksina vain kaikkein parhaimpia dokumentteja, koska osittain relevantteja dokumentteja voi olla kymmeniä tuhansia. Brin & Page pitävät relevanttien hakutulosten määrää ensimmäisten hakutulosten joukossa jopa tärkeämpänä kuin relevanttien hakutulosten kokonaismäärää. Tämä on loogista, koska sen hakutulosten joukon laadulla, jota käyttäjä ei katso, ei myöskään ole käyttäjän näkökulmasta mitään väliä. Esimerkiksi Google antaa n. 4 300 000 000 hakutulosta hakusa-

nalla “general”. Käyttäjä ei todennäköisesti välitä minkä laatuisia hakutulokset ovat ensimmäisten kymmenien jälkeen. Tietysti voidaan esittää kysymys siitä, kuinka tärkeitä nämä hakutulokset ovat tietokoneille.

Hakukoneen yksi keskeisimmistä ominaisuuksista on indeksin avulla löydettyjen hakutulosten järjestäminen niiden laadun mukaan. Koska hakutulosten laatu on yksi suurimpia kilpailuvaltteja hakukonemarkkinoilla, hakutulosten järjestämiseen käytetyistä algoritmeista paljastetaan harvoin yksityiskohtaisia tietoa. Hakutulosten optimoinnista ja suoranaisesta manipuloinnista on tullut niin kannattavaa, että osapuolten välille syntyy väistämättä eturistiriitoja, mikä antaa lisäsyyn pitää hakukoneiden järjestämisalgoritmit pitkälti salaisina ja alati muuttuvina. Järjestysalgoritmien tuntemisesta onkin syntynyt aivan oma ammattitaitonsa – hakukoneoptimointi. (Killoran 2013) Tiedetään kuitenkin, että Web-dokumenttien HTML-rakennetta, Webin linkkirakennetta ja dataa käyttäjien toiminnasta hyödynnetään (Brin & Page 1998).

Googlen PageRank on varmasti tunnetuin ja menestynein hakutulosten järjestys algoritmi. Myös Googlen algoritmin tarkat yksityiskohdat ovat salaisia, mutta tiedetään, että osittain se hyödyntää edelleen Brin & Pagen (1998) esittelemää alkuperäistä algoritmia. Alkuperäinen ajatus PageRank-algoritmin takana on, että Web-sivun tärkeys voidaan määrittää sen pohjalta kuinka monet muut sivut linkittävät siihen ja kuinka tärkeitä nämä muut sivut ovat. Myöhemmin Google on kuitenkin lisännyt algoritmiinsa useita uusia tekijöitä. Googlen hakutulosten järjestykseen sanotaan vaikuttavan yli 200 tekijää, joista PageRank on vain yksi (Killoran 2013). PageRankin lisäksi muita tekijöitä ovat muun muassa suosio sosiaalisessa mediassa, sivun kävijämäärät, välitön poistumisprosentti sivulta ja sivun latausnopeus (Dean 12.5.2015). Näiden lisäksi käyttäjä voi itse rajata Googlen hakutuloksia kielen, ajankohdan ja sijainnin perusteella.

Osa hakukoneista, esimerkiksi Hakia (julkinen palvelu lopetettu), Kosmix (toiminta lopetettu), DuckDuckGo, Bing, Kngine (http://www.kngine.com/), Factbites (http://www.factbites.com/), Lexxe (http://www.lexxe.com/), Cluzz (http://www.cluuz.com/), SenseBot (http://www.sensebot.net/), Swoogle (http://swoogle.umbc.edu/), Watson (http://watson.kmi.open.ac.uk/), Falcons (http://ws.nju.edu.cn/falcons/), Exalead (https://www.exalead.com/search/), Powerset[2], pyrkivät hyödyntämään semanttista analyysiä parempien hakutulosten tuottamiseen. Semantiikkaa hyödyntävät hakukoneet pyrkivät paremmin ymmärtämään hakusanojen tarkoituksen ja etsimään hakutuloksia laajemmin aiheeseen liittyen toisin kuin perinteiset avainsanahakukoneet, jotka etsivät hakutuloksia vain annetuilla avainsanoilla. Semanttiset hakukoneet hyödyntävät ontologiaa, luonnollisen kielen ymmärrystä, päättelyä ja kontekstin ymmärrystä tukevia teknologioita. (Khan et al. 2014, Lai et al. .2011.)

[2] Powerset on nykyään osa Microsoftia. (Search Engine Land 2008.)

3.3 Paikkapohjainen Web-haku

Web-dokumenttien sisältämästä tiedosta suuri osa liittyy johonkin tiettyyn maantieteelliseen sijaintiin. Mobiililaitteiden yleistyessä mobiilihauista on tullut varsin yleisiä. Mobiilikäyttäjää kiinnostaa usein "tässä ja nyt" tyyppinen tieto, mikä on tehnyt maantieteellisestä sijainnista keskeisen tekijän tiedonhaussa (Ahlers & Doll 2007). Zhou et al. toteaa, että vaikka avainsanahakukoneissa voi käyttää esimerkiksi paikkojen tai maamerkkien nimiä muiden avainsanojen lisäksi, usein se kuitenkin johtaa epärelevantteihin hakutuloksiin, koska hakukone ei välttämättä ymmärrä hakusanoja maantieteellisinä sijainteina tai paikannimeä on käytetty Webissä muuhun tarkoitukseen. Myös laadukkaita hakutuloksia karsiutuu pois vain sen takia, että ne eivät sisältäneet juuri kyseisiä paikannimiä, vaikka liittyvätkin kyseiseen alueeseen (2001). Toisaalta Ahlers toteaa, että nykyaikaiset avainsanahakukoneet pyrkivät ymmärtämään Web-dokumenttien paikkatietoja jossain määrin ja painottamaan hakutuloksia tämän perusteella, jos sattuvat ymmärtämään haun ja hakutulosten maantieteellisen luonteen (Ahlers 2012).

Web-dokumenttien paikkatietoa hyödyntämällä voidaan toteuttaa hakukone, joka palauttaa hakutuloksia sijainnin perusteella. Yhä isompi osa Web-hauista tehdään mobiililaitteilla. Googlen tietojen mukaan hauista suurin osa tehdään jo mobiililaitteilla (Dischler 2015). Mobiililaitteilla tehtyjen hakujen aiheet ovat pitkälti vastaavia kuin muilla laitteilla tehdyillä hauilla, mutta ne painottuvat selkeämmin paikallisiin hakutuloksiin (Ahlers 2012 s. 55). Tämän jo pitkään jatkuneen trendin myötä paikkapohjaisia Web-hakuja tukevan hakukoneen toteuttaminen on herättänyt mielenkiintoa sekä yrityksissä että tutkijayhteisössä (Ahlers 2012 s. 49; Tabarcea et al. 2010). Useat kaupalliset hakukoneet (esim. Google local ja Yahoo local) pyrkivät tarjoamaan paikallisia hakutuloksia, mutta niiden hakutulokset rajoittuvat pääsääntöisesti vain yrityslistauksiin, jotka tuotetaan kaupallisten yritystietokantojen ja käyttäjien tuottaman datan perusteella (Dlugolinsky et al. 2010; Zhou et al. 2005; Vänskä 2004). Tässä työssä mielenkiinnon kohteena on sisällön etsiminen Webistä yleisellä tasolla rajoittumatta mihinkään tietyntyyppiseen tietoon.

Paikkapohjaiset Web-haut voidaan myös jakaa muutamaan eri tyyppiin. Osassa hauista käyttäjä yrittää etsiä tietoa jo tuntemastaan paikasta kun taas osassa hauista käyttäjä etsii tietoa ennalta tuntemattomasta paikasta tietyillä hakukriteereillä. Näiden lisäksi voidaan vielä eritellä niin sanottu tutkivahaku, jossa käyttäjä etsii tietoa ympäristöstään ilman, että etsisi mitään tiettyä tietoa. (Ahlers 2012 s. 54) Tämän työn keskiössä on nimenomaisesti tutkivahaku.

3.3.1 Paikkatiedon kerääminen

Hakukonetta, joka sallii Web-sisällön etsimisen sijainnin perusteella, kutsutaan tässä työssä paikallishakukoneeksi. Paikallishakukoneella täytyy olla käsitys Web-dokumenttien ja maantieteellisten sijaintien yhteyksistä, jotta se voi tuottaa hakutuloksia. Web-do-

kumentit sisältävät usein viittauksia maantieteelliseen sijaintiin, mutta tieto on useimmiten luonnollisen kielen muodossa ja tästä syystä vaikeasti kerättävissä (Loglisci et al. 2012; Ahlers & Boll 2007). Viittausta maantieteelliseen sijaintiin kutsutaan tässä työssä paikkatiedoksi. Web-dokumentin sisältämä paikkatieto voi olla esimerkiksi osoitteen, paikan nimen tai koordinaattien muodossa.

Paikallishakukoneiden vaatimaa tietoa Web-dokumenttien ja maantieteellisten sijaintien välillä on yritetty kerätä useilla eri tavoilla. Ahlersin (2012) paikallishakukoneanalyysin ja omakohtaisen kokemukseni perusteella voin todeta, että paikallishakukonetta, joka pystyisi laajamittaisesti ja tehokkaasti löytämään Web-sisältöä sijainnin perusteella, ei näyttäisi olevan olemassa. Paikkatiedon keruu tavat ovat moninaiset, mutta selkeästi vielä riittämättömät.

Tutkimusta on tehty paljon paikkatiedon eristämisestä suoraan Webistä. Web-dokumenttien sisältämä paikkatieto voi olla eksplisiittisessä tai implisiittisessä muodossa (Ahlers & Boll 2008). Implisiittisellä muodolla viitataan dokumentin sisällöstä pääteltävissä olevaan paikkatietoon. Paikkatieto voi olla dokumentin sisällössä esimerkiksi osoitteena tai paikannimenä. Dlugolinskyn ja kumppaneiden suorittamassa kokeessa 14.34 % tutkituista 408 096 Web-dokumentista sisälsi osoitetietoja (2010). Vänskä (2004) tutki 24 000 suomalaista Web-sivua, joista 35 % sisälsi paikkatietoelementtejä (paikannimiä, kadunnimiä, postinumeroita, puhelinnumeroita tai osoite-elementtejä).

Useimmiten Web-dokumentin paikkatieto on nimenomaan implisiittisessä muodossa (Tabarcea et al. 2010; Vänskä 2004). Web-dokumenttien paikkatiedon eksplisiittiseen määrittämiseen on esitetty muun muassa HTML-standardin mukaisia `meta`-tageja. `Meta`-tagien avulla paikkatieto voidaan määrittää HTML-dokumenttiin esimerkiksi seuraavasti:

```
<meta name="ICBM" content="50.38734, -43.9874323">
```

ICBM-tagin tagin `content` attribuutin arvoksi asetetaan WGS-84 koordinaattijärjestelmän mukaiset pituus- ja leveyskoordinaatit pilkulla (`,`) erotettuna. Desimaalierottimena käytetään pistettä. Desimaalien määrää ei ole rajoitettu. Leveys- ja pituuskoordinaattien lisäksi tagiin voi määrittää myös korkeuden metreinä seuraavasti:

```
<meta name="ICBM" content="50.38734, -43.9874323,230">
```

Korkeus (esimerkissä 230 m) ilmoitetaan siis pilkulla erotettuna koordinaattien viimeisenä osana.

Vastaavasti voi käyttää myös geo `meta`-tageja seuraavasti:

```
<meta name="geo.position" content="50.38734;-43.9873">
<meta name="geo.placename" content="Muonio, Suomi">
<meta name="geo.region" content="FI-10">
```

`Geo.position`-tagin `content` attribuutti on muutoin vastaava kuin `ICBM`-tagin sisältö, mutta koordinaatit pitää erotella puolipisteellä (`;`) pilkun sijaan. `Geo.placename` sisältää paikannimen tekstuaalisessa muodossa. Kentän arvolle ei varsinaisesti ole mitään rajoituksia, mutta on tietysti suositeltavaa, että arvoksi annetaan maantieteellisesti merkityksellinen nimi. `Geo.region` sisältää maakoodin ja mahdollisen aluekoodin väliviivalla (`-`) erotettuna. Maakoodin tulee olla ISO 3166-1-standardin mukainen maakoodi ja aluekoodin tulee olla ISO 3166-2-standardin mukainen aluekoodi. (Daviel & Kaegi 2007.)

Edellä mainittujen `meta`-tagien standardointiprosessi näyttäisi kuitenkin pysähtyneen Daviel & Kaegin ehdotelman aikarajan mentyä umpeen 2008 (IETF 12.5.2015). Tagien laajamittaisesta käytöstä ei myöskään ole havaintoa. Vänskä (2004) kävi tutkimuksessaan läpi n. 24 000 suomalaista Web-sivua, joista yksikään ei sisältänyt geo-tageja. Dlugolinskin (2010) mukaan myöskään muun muassa Google Maps ei hyödynnä löytämiään paikkatieto `meta`-tageja. Ahlers ja Boll (2008) toteaa strukturoidun paikkatiedon olevan erittäin harvinaista Web-dokumenteissa. Toisaalta myöhemmin Ahlers (2012) toteaa, että uusimmat paikallishakukoneet käyttävät jossain määrin Web-dokumenteista löytyvää strukturoitua paikkatietoa mainiten esimerkkinä mikroformaatit ja W3C:n standardoiman RDF mallin. Tässä nähdään, ensinnäkin se, että kaupallisten hakukoneiden toteutusyksityiskohdat ovat suurelta osin pimennossa, mutta voidaan toisaalta myös ehkä päätellä, että paikkatiedon semanttinen merkkaus on jossain määrin lisääntymään päin.

Myös HTML:n `<address>`-tagia voidaan hyödyntää osoitteiden semanttiseen merkkaamiseen. Tabarcea et al. (2010) mukaan kuitenkaan edes `<address>`-tagin käyttö ei ole yleistä Web-sivustoilla.

Eksplisiittisen tiedon puutteesta huolimatta Webistä voidaan eristää huomattava määrä paikkatietoa (Ahlers 2012). Web-dokumenttien implisiittisen paikkatiedon hyödyntäminen paikallishakukoneessa vaatii, että ensin paikkatietoa esittävät entiteetit (osoitteet, paikannimet, kaupunkien nimet yms.) tunnistetaan ja eristetään dokumentista. Tekstuaalisen paikkatiedon eristämisen jälkeen se täytyy vielä saattaa eksaktiin indeksoitavaan muotoon, mikä tapahtuu pääsääntöisesti geokoodaamalla. Geokoodauksessa tekstuaalisessa muodossa olevalle paikkatiedolle määritetään leveys- ja pituuskoordinaatit. Esimerkiksi Google tarjoaa geokoodausta palveluna. Koordinaattien avulla paikkatieto voidaan indeksoida tehokkaasti tietokantaan. (Dlugolinsky et al. 2010.)

Kirjallisuudessa (Fränti et al. 2010; Ahlers & Boll 2008; Vänskä 2004) on osoitettu, että osoitemuotoisen paikkatiedon eristäminen Web-dokumenteista on täysin mahdollista. Kussakin edellä mainituista tutkimuksessa osoitteiden eristäminen vaati kuitenkin maakohtaista tuntemusta osoitteiden tyypistä ja osassa maakohtaista paikkatietorekisteriä, jotta löydettyjen osoitteiden laatua ja oikeellisuutta pystyttiin arvioimaan.

Paikkatietoa sisältäviä rekistereitä (engl. gazetteer) on kerätty muun muassa valtioiden yritysten ja yhteisöjen toimesta. Paikkatietorekistereillä on pyritty keräämään sekä liiketoiminnalle että valtiolle keskeistä paikkatietoa. Tietoa on kerätty hyvin monipuolisesti erilaisista maantieteellisistä kohteista, kuten teistä, vesistöistä, metsistä, kaupungeista, rakennuksista, maanosista ja erilaisista kohteista (Vänskä 2004). Kohdetieto (engl. Point Of Interest – POI) on maantieteellisenä pisteenä määritelty sijainti, johon on liitetty kiinnostavaa tietoa. Osa näistä paikkatietorekistereistä on julkisesti kaikkien saatavilla, etenkin useimmat julkisen vallan tuottamat paikkatietorekisterit. Yritysten keräämät paikkatietorekisterit ovat pääsääntöisesti joko yritysten sisäiseen käyttöön tarkoitettuja tai käytettävissä maksua vastaan.

Vielä vähän aikaa sitten useimmat paikallishakukoneet perustuivat nimenomaan paikkatietorekistereistä koostettuun tietoon. Esimerkiksi Google Maps ja YellowPages.com yhdistävät yritystietokannoista ja paikkatietorekistereistä löytyvää tietoa. Nykyään yhä useammat palvelut ja laitteet lisäävät luomansa sisällön metatietoihin paikkatietoa kerryttäen itselleen paikkatietotietokantaa. Esimerkiksi Wikipedia , Flickr (https://www.flickr.com/) ja useimmat sosiaalisen median palvelut pyrkivät lisäämään luomaansa sisältöön paikkatietoa, jotta voisivat tarjota käyttäjille paikkatietoa hyödyntäviä ominaisuuksia. (Ahlers 2012.)

Yhä useammin paikallishakukoneet kuitenkin käyttävät kuratoitujen tietokantojen sijaan Webistä kerättyä paikkatietoja. Vaikka kuratoidut tietokannat voivatkin tarjota tarkkaa laadukasta paikkatietoa, niin niiden laajuus ja tiedon syvyys ei kuitenkaan kilpaile suoraan Webistä kerätyn paikkatiedon kanssa. (Ahlers 2012) Kuratoidut paikkatietotietokannat toimivat erikoistuneen paikallishakukoneen perustana, mutta yleiskäyttöinen paikallishakukone vaatii perustaksi laajemmin Webin tietosisältöä analysoivan pohjan.

Eräs mielenkiintoinen tapa liittää Web-dokumentteihin paikkatietoa on analysoida käyttäjien tekemiä hakuja. Käyttäjien tekemistä hauista jää lokitietoja joiden perusteella voidaan selvittää, mistä maantieteellisestä sijainnista haku tehtiin. Analysoimalla kaikki tiettyyn hakuun liittyvät maantieteelliset sijainnit, voidaan huomata, että haut voivat olla tiukasti tiettyyn paikkaan sidottuja tai hajautua laajalle alueelle kuitenkin niin, että haulla on maantieteellinen keskipiste tai useampia keskittymiä. Hakujen maantieteellinen painopiste voi myös liikkua ajan myötä. (Backstrom et al. 2008.)

Analysoimalla hakusijaintien maantieteellistä jakaumaa ja keskipisteitä voidaan esimerkiksi hakukoneiden mainontaa kohdentaa paremmin. Tietoa voidaan myös hyödyntää hakutulosten painotuksessa. Esimerkiksi tiettyjä uutisia voidaan painottaa hakutuloksissa niillä alueilla, missä kyseistä uutista on haettu paljon. (Backstrom et al. 2008.)

3.3.2 Paikallishakukoneet

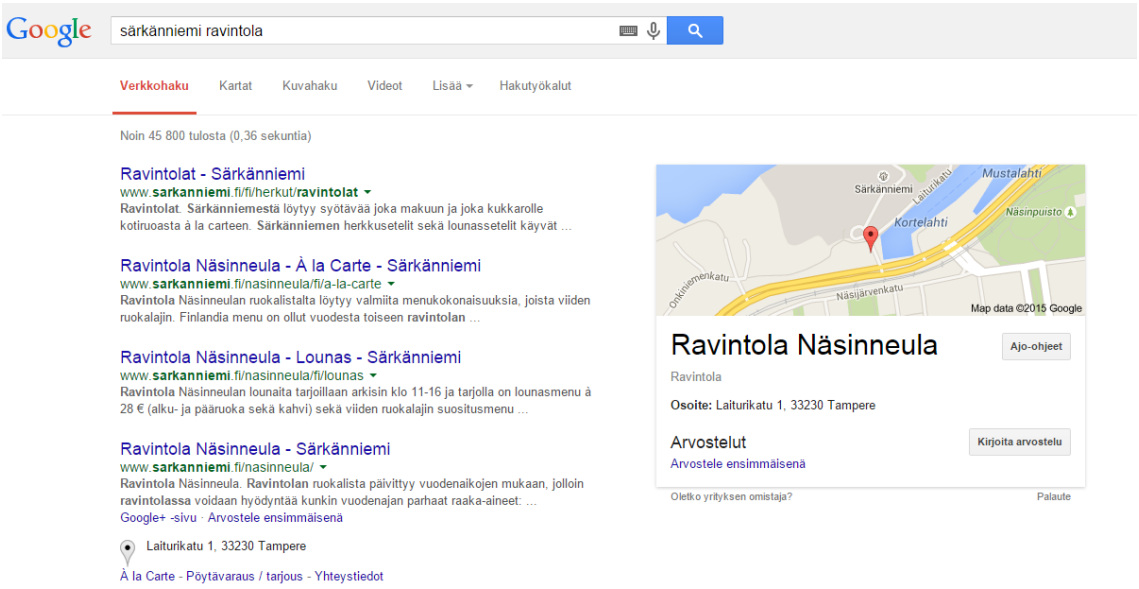
Paikallishakujen tuottamat hakutulokset on tärkeää näyttää käyttäjälle siten, että hakutulosten maantieteellinen sijainti ja tarvittaessa käyttäjän oma sijainti on selkeästi esitetty (Zhou et al. 2001). Tyypillisesti paikallishakukoneiden hakutulokset näytetään karttapohjalla, mutta myös muita vaihtoehtoja on. Esimerkiksi Nokian City Lens ja Googlen Google Glasses näyttävät hakutuloksia lisätyn todellisuuden avulla.

Kuvassa 2 on havainnollistettu, kuinka hakutuloksia voidaan näyttää suoraan puhelimen kameran kuvassa. Nokia City Lensin lähestymistapaa kutsutaan lisätyksi todellisuudeksi (engl. augmented reality).

Google Places on Googlen palvelu, joka listaa paikallisia yrityksiä ja erilaisia kohdetietoja. Yrittäjät voivat itse antaa Googlelle yrityksensä tiedot. Google pyrkii varmistamaan tietojen lähettäjän henkilöllisyyden puhelimitse, tekstiviestillä tai kirjeellä (Google Support 12.5.2015). Google Places palvelussa hyödynnetään tietysti muutakin Googlen keräämää dataa, mutta tarkempaa tietoa Googlen tavoista kerätä ja käsitellä paikkatietodataa on vaikea löytää[3]. Ymmärrettävästi tällainen tieto halutaan pitää pitkälti yrityssalaisuutena. Yleisesti on kuitenkin tiedossa, että Google kerää dataa aktiivisesti useista saatavilla olevista lähteistä.

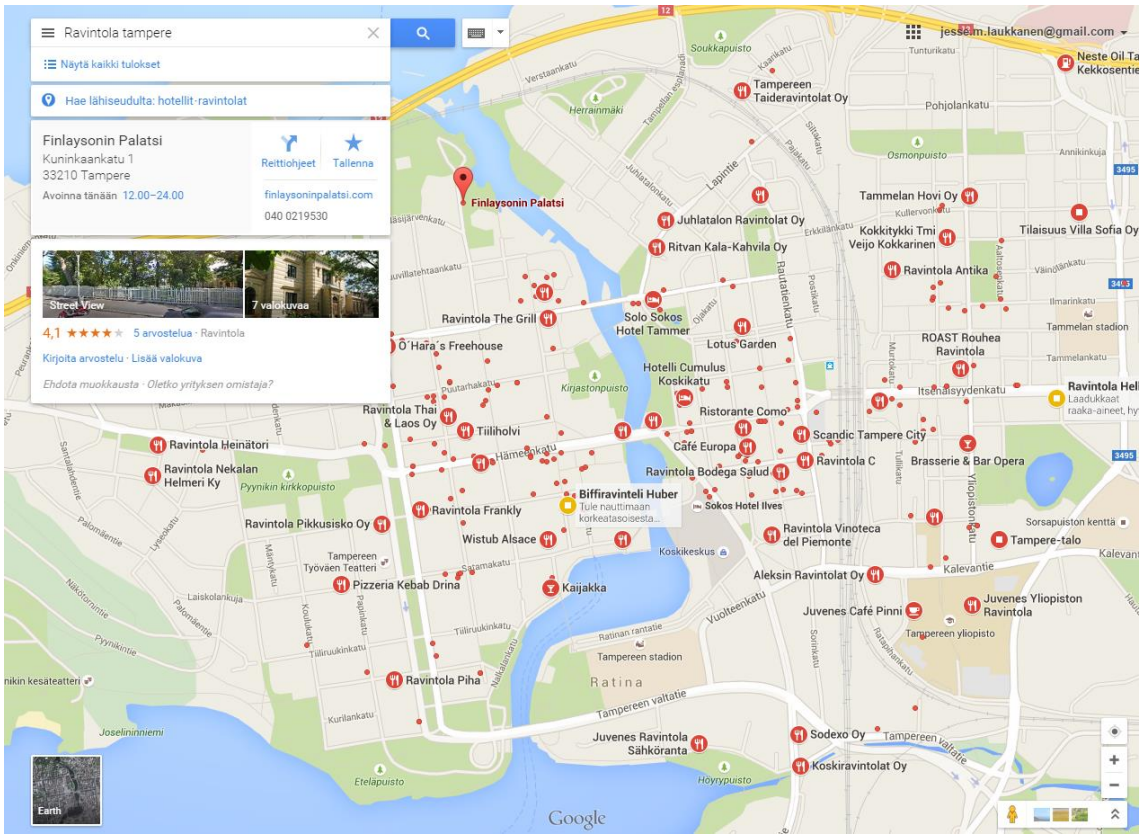
[3] Googlen datalähteistä on esitetty valistuneita arvauksia useiden ammattilaisten toimesta. Esimerkiksi Mihm (9.5.2015) on esittänyt hyvän yhteenvedon aiheesta.

Kuva 2. *Nokia City Lens havainnekuva.*



Kuva 3. *Google Places hakutulos normaalien hakutulosten yhteydessä.*

Google Places hakutuloksia näytetään sekä normaalien hakujen yhteydessä sivun oikealla palstalla että Google Maps karttahakujen yhteydessä. Kuvassa 3 havainnollistetaan Google Places hakutuloksen näyttötapa Googlen yleishaussa.



Kuva 4. *Google Maps -karttahaku.*

Google Maps karttahaussa näytetään Google Places hakutuloksia sekä karttamerkkeinä että tietokortteina vasemmassa yläreunassa (katso kuva 4). Google Place hakutuloksia hyödynnetään myös Google Glass laitteessa, joka näyttää hakutuloksia käyttäjän näkökentässä lisätyn todellisuuden periaatteita noudattaen. Lopulta hakukäyttöliittymä voi olla lähes mitä tahansa. Paikallisia hakutuloksia voidaan näyttää esimerkiksi Google Earth palvelussa, navigaattoreissa, mobiilisovelluksissa, kehittyvissä maissa tekstiviestejä ja jopa puhelinta käytetään käyttöliittymänä (Ahlers 2012).

Google Place hakutulokset tarjoavat käyttäjälle tietokortin hakutuloksen kohteesta ja mahdollisen linkin hakutulokseen liittyvään Web-sivustoon. Tietokorteissa listataan yrityksen keskeisimmät tiedot kuten nimi, osoite, puhelinnumero, aukioloajat, WWW-osoite ja muutama kuva yrityksestä. Tietoja listataan siis sikäli, kun ne ovat Googlella tiedossa. Esimerkiksi aukioloaikoja ei löydy läheskään kaikista hakutuloksista. Google Places ratkaisu pyrkii tarjoamaan käyttäjälle oleellisen tiedon ilman, että käyttäjän tarvitsisi lukea varsinaista hakutulokseen liittyvää Web-sivustoa. Tällainen lähestymistapa on varsin perusteltu, kun pyritään löytämään käyttäjälle oleellinen tieto mahdollisimman tehokkaasti.

Ensimmäiset todelliset yritykset luoda paikallishakukoneita perustuivat osoitteiden parsimiseen Web-dokumenteista. Tämä lähestymistapa ei kuitenkaan tuottanut tulosta ja siirryttiin käyttämään pelkästään yritystietokantojen tarjoamia tietoja. Nykyaikaiset paikallishakukoneet ovat hybridejä, sillä ne keräävät dataa useista eri lähteistä. Dataa kerätään

yritystietokannoista, Webistä (osoitteiden muodossa), yritysten omistajilta ja lopulta käyttäjiltä kerätään päivityksiä ja korjauksia dataan joukkoistamalla. Data lähteitä käsitellään yksilöllisesti ja niistä saadun datan luotettavuutta painotetaan lähteen luotettavuuden perusteella. Esimerkiksi keltaisten sivujen tietokannasta tuleva yritystieto todennäköisesti näytetään hakutuloksissa. Jos samaan yritykseen pystytään vielä liittämään lisätietoa Webistä löytyneen datan avulla, hakutulosta saatetaan painottaa enemmän. Toisaalta pelkkä Webistä löytynyt tieto ei välttämättä johda siihen, että yritys näytettäisiin hakutuloksissa. (Ahlers 2012)

4. YHTEISÖLLINEN PAIKALLISHAKUKONE

Tämän työn rinnalla toteutettiin esimerkkitoteutukseen pohjautuva hakukone prototyyppi Glinks[4]. Prototyypin nimi tulee sanoista Geocoded links. Prototyypin tavoitteena on kerätä käytännön kokemusta Web-dokumentteihin liittyvän paikkatiedon keräämisestä joukkoistamalla sekä kerätyn tiedon hyödyllisyydestä. Prototyyppi on hakukone, joka tarjoaa rekisteröityneille käyttäjille mahdollisuuden tuottaa ja ylläpitää hakukoneen tarvitsemaa paikkatietoa sekä kaikille käyttäjille mahdollisuuden etsiä Web-dokumentteja sijainnin perusteella.

Tässä luvussa kuvataan esimerkkitoteutus yhteisöllisestä joukkoistamiseen pohjautuvasta paikallishakukoneesta. Esimerkkitoteutus kuvaa kuinka nykyisillä Web-teknologioilla voidaan toteuttaa paikallishakukone, joka tukeutuu pääasiallisesti käyttäjäyhteisön tuottamaan ja ylläpitämään paikkatietoon.

4.1 Toiminnallinen kuvaus

Yhteisöllisen paikallishakukoneen päätavoitteena on tarjota käyttäjille tapa etsiä Web-dokumentteja maantieteellisen sijainnin perusteella. Hakuparametrina voidaan käyttää esimerkiksi käyttäjän omaa maantieteellistä sijaintia, joka saadaan tarvittaessa nykyaikaisissa mobiililaitteissa tuetun GPS-paikannusominaisuuden avulla.

Yhteisöllinen paikallishakukone perustuu käyttäjien luomiin geokoodattuihin linkkeihin eli tietueisiin, jotka sisältävät URL-viittauksen Web-dokumenttiin ja viittauksen maantieteelliseen sijaintiin leveys- ja pituuskoordinaatteina. Yksi tällainen tietue mallintaa siis yhden Web-dokumentin ja yhden maantieteellisen sijainnin yhteyttä. Tietueiden perusteella käyttäjälle voidaan näyttää hakutuloksia hakuparametrina käytetyn sijainnin läheltä. Tietueen sisältämä paikkatieto riittää paikallisten hakutulosten listaamiseen sekä hakutuloksen ja hakuparametrina käytetyn sijainnin välisen etäisyyden laskemiseen. Etäisyyttä voidaan hyödyntää hakutulosten järjestämiseen.

Yhteisöllisen paikallishakukoneen hakutulokset ovat käytännössä vastaavia viittauksia Web-dokumentteihin kuin perinteisten avainsana hakukoneiden hakutulokset. Suurin ero on, että hakutulokset määräytyvät avainsanojen sijaan hakuparametrina käytetyn sijainnin mukaan. Paikallishakukoneeseen voi tietysti yhdistää tarvittaessa avainsanahaun ominaisuuksia. Esimerkiksi hakutuloksia voisi järjestää tai rajata sijainnin lisäksi hakusanojen perusteella. Hakusanojen käyttäminen vaatii kuitenkin Web-dokumenttien sisällön tarkkaa analysointia ja tehokasta indeksointia, joka itsessään on varsin laaja aihealue kuten

[4] Prototyyppi on kokeiltavissa (8.5.2015) osoitteessa: https://glinks.io

muun muassa Sergey Brin ja Larry Page (1998) havainnollistavat työssään. Tästä syystä hakusanojen hyödyntäminen paikallishakukoneessa on jätetty tämän työn ulkopuolelle.

Paikallishakukoneen keskeisin tehtävä on kerätä paikkatietodataa, jotta paikkaan sidottuja hakutuloksia voidaan tarjota hakukoneen käyttäjille. Kuten aikaisemmin on todettu muun muassa Vänskän (2004) toimesta, juuri tämän Web-dokumentteihin liittyvän paikkatiedon tehokas kerääminen on haasteellista. Joukkoistaminen on kuitenkin yksi mahdollinen tapa ratkaista tämä ongelma laajamittaisesti. Wikipedia, Stack Overflow ja monet muut Webin palveluista ovat hyviä esimerkkejä siitä mihin laajuuteen onnistuneella joukkoistamisella pystytään. Lisäksi joukkoistamalla päädytään mallintamaan ihmisten käsitystä Web-dokumenttien sisällön ja maantieteellisen sijainnin välillä. Näin myös sellaisiin Web-dokumentteihin, joihin ei liity helposti ohjelmallisesti eristettävää paikkatietoa, pystytään liittämään tarkkaa paikkatietoa.

4.1.1 Paikkatiedon kerääminen joukkoistamalla

Sosiaalinen paikallishakukone antaa rekisteröityneistä käyttäjistä muodostuneelle yhteisölle vallan luoda ja ylläpitää Web-dokumentteihin liitettyä paikkatietoa. Yhteisö edustaa paikallishakukoneen aktiivisia käyttäjiä. Yhteisöllä on suora mahdollisuus vaikuttaa paikkatiedon luontiin ja ylläpitoon. Tästä seuraa se, että hakukoneeseen kertyvä paikkatieto kuvaa nimenomaan yhteisön näkemystä Web-dokumenttien ja maantieteellisten sijaintien yhteydestä.

Hakutulokset pohjautuvat yhteisön kollektiiviseen näkemykseen Web-dokumenttien ja maantieteellisten sijaintien yhteydestä ja tästä syytä hakutulosten laatu on täysin sidoksissa siihen kuinka hyvin yhteisö toimii. Käyttäjät liittävät Web-dokumentteja sijainteihin, joihin he itse kokevat niiden kuuluvan. Yhteisön tehtävä on valvoa kerääntyvän paikkatiedon laatua. Yhteisö on siis itse suoraan vastuussa hakutulosten laadusta.

Kuka tahansa voi rekisteröityä paikallishakukoneen käyttäjäksi ja saada oikeuden vaikuttaa hakutulosten sisältöön ja laatuun. Rekisteröityneiden käyttäjien täytyy kirjautua järjestelmään sisälle voidakseen osallistua hakukoneen yhteisölliseen toimintaan. Kirjautumisella varmistetaan, että käyttäjän toimia yhteisössä voidaan seurata ja arvioida.

Kuten muun muassa Bishop (2013) on kuvannut työssään, verkkoyhteisöissä ilmenee aina myös ei-toivottua käyttäytymistä. Sosiaalisen paikallishakukoneen yksi keskeinen osa on mainejärjestelmä. Mainejärjestelmän vastuulla on mallintaa yhteisön jäsenten mainetta yhteisössä, jotta käyttäjien luoman sisällön laatua voidaan arvioida tarkemmin.

Mainejärjestelmän suunnittelussa noudatetaan Alnemrin ja Meinelin (2011) esittämiä suunnitteluperiaatteita. Hakukoneen keskeisimmät entiteetit ovat hakutulos ja käyttäjä. Kullakin käyttäjällä on numeraalisesti ilmaistavissa oleva maine, isompi numeraalinen arvo kuvaa parempaa mainetta ja pienempi arvo huonompaa mainetta. Mainejärjestelmän

vastuulla on myös antaa yhteisölle työkalut, joiden avulla yksilöt voivat vaikuttaa toisten yksilöiden maineeseen.

Maineeseen voidaan vaikuttaa arvioimalla toisten yksittäisiä toimia äänestämällä niiden puolesta ja vastaan. Äänestämällä toimen puolesta käyttäjä vahvistaa toimen tehneen mainetta yhteisössä. Äänestämällä tointa vastaan käyttäjä vastavuoroisesti heikentää toimen tehneen mainetta. Äänestämällä yhteisö määrittelee kollektiivisesti yksilöiden maineen. Käyttäjien maine on julkinen tieto, joka näytetään kaikille muille käyttäjille. Näin yhteisössä pyritään mallintamaan luottamusta, minkä Abdul-Rahman & Hailes (2000) totesivat olevan tärkeää.

Yksilöiden mainetta käytetään vallan jakamisen perusteena yhteisössä. Valtaa rajoitetaan käyttöoikeuksilla. Mitä parempi maine käyttäjillä on, sitä enemmän heidän toimiin voidaan luottaa ja sitä enemmän heille annetaan käyttöoikeuksia. Luottamus tuo siis valtaa. Kuten reaalimaailmassa niin myös verkkoyhteisössä luottamus täytyy ansaita luottamusta herättävällä käytöksellä. Luottamus ei myöskään ole pysyväistä vaan muuttuu ajan myötä. Julkinen maine, mahdollisuus vaikuttaa käyttäjien maineeseen äänestämällä ja rajoittaa näin käyttäjien oikeuksia tarjoavat yhteisölle vähimmäiskeinot valvoa ja sanktioida yhteisön jäsenien toimintaa, minkä Kollock & Smith (1996) totesivat olevan yhteistä kaikille menestyneille verkkoyhteisöille.

Käyttäjäyhteisö voi luoda, muokata, poistaa ja arvioida hakukoneeseen kertyvää paikkatietoa. Kukin yhteisön jäsen voi osallistua toimintaan oikeuksiensa mukaisesti. Käyttäjän oikeudet osallistua yhteisön toimintaan riippuvat suoraan käyttäjän maineesta. Hyvämaineisilla käyttäjillä on enemmän oikeuksia järjestelmässä kuin huonomaineisilla. Uusien käyttäjien maine on aluksi olematon ja hyvä maine ansaitaan käyttämällä vähäisiä oikeuksia yhteisön hyväksi.

Hakukone antaa käyttäjille mahdollisuuden luoda hakukoneen vaatimaa paikkatietoa valitsemalla sijainti kartasta tai syöttämällä sijainti tekstikenttään. Sijaintiin liittyvään Web-dokumenttiin viitataan URL-osoitteella. Järjestelmä hakee annettua URL-osoitetta vastaavan dokumentin ja raapii dokumentista metatietoja, jotta hakutuloksissa voidaan näyttää lyhyt yhteenveto kyseisestä dokumentista. Metatietoraapija hakee dokumentin favikonin, otsikon ja lyhyen kuvauksen. Käyttäjällä on mahdollisuus muokata järjestelmän ehdottamaa otsikkoa ja kuvausta.

Tämä lähestymistapa sallii sen, että yksittäinen Web-dokumentti voi liittyä useaan sijaintiin. Riittää, että joku yhteisön jäsenistä kokee dokumentin liittyvän sijaintiin, johon sitä ei ole vielä liitetty ja lisää tätä varten järjestelmään uuden geokoodatun linkin.

4.1.2 Hakuominaisuudet

Hakukone antaa käyttäjälle mahdollisuuden etsiä Web-dokumentteja sijainnin perusteella. Sijaintina voidaan käyttää käyttäjän omaa sijaintia, käyttäjän antamaa tekstimuotoista sijaintia tai käyttäjän karttaan kohdistamaa sijainti. Sijainnin lisäksi hakukone antaa mahdollisuuden rajata haku tietylle alueelle sijainnin ympärillä. Käyttöliittymän karttaa zoomaamalla käyttäjä voi rajata haun laajuutta. Kartta voidaan esimerkiksi kohdistaa tiettyyn kaupunkiin ja rajata karttaan vain ydinkeskustan alue.

Hakuparametrina käytetään käyttöliittymässä näytettävän kartan rajaamaa aluetta. Kartta on kohdistettu ennen hakua joko käyttäjän antaman sijainnin perusteella tai HTML5-paikannuksen avulla. Varsinainen haku tehdään karttaan rajatun suorakulmaisen karttanäkymän perusteella. Tätä karttaan rajattua aluetta kutsutaan rajauslaatikoksi. Rajauslaatikon keskustassa sijaitsee haun keskiö. Palvelinohjelmiston tehtävänä on tuottaa rajauslaatikon mukainen hakutulosjoukko.

Hakuominaisuudet tukevat hakujen tekemistä etenkin mobiililaitteilla, koska oletettavasti yleisimmät paikallishakukoneen käyttötapaukset liittyvät tilanteisiin, joissa käyttäjä on liikkeellä ja käyttää jonkinlaista mobiililaitetta tiedon etsimiseen omasta ympäristöstään.

Kuva 5. *Favikonilla rikastettu karttamerkki.*

Hakutulokset visualisoidaan karttakäyttöliittymässä karttamerkkeinä. Karttamerkeissä hyödynnetään sivuston favikonia (katso kuva 5). Favikonia hyödyntämällä karttamerkeistä saadaan yksilöllisiä ja tunnistettavia. Karttamerkkien lisäksi hakutulokset listataan tekstimuotoisessa listanäkymässä. Listanäkymässä käyttäjälle näytetään hakutuloksen otsikko, lyhyt yhteenveto hakutuloksessa viitatusta Web-dokumentista sekä etäisyys hakutuloksen ja hakuparametrina käytetyn sijainnin välillä.

Kartalla visualisoidaan myös käyttäjän oma sijainti, jos se on saatavilla. Kartalla visualisoidaan myös arvio käyttäjän paikannuksen tarkkuudesta. Käyttäjän oma sijainti ja paikannuksen tarkkuus on havainnollistettu kuvassa 6.

Paikallishakukone hyödyntää W3C:n määrittelemää HTML5-geopaikannus rajapintaa käyttäjän paikantamiseen. Rajapinnan avulla selaimelta tiedustellaan käyttäjän sijaintia, mikäli käyttäjältä saadaan tähän ensin lupa. Lupaa paikannuksen tekemiseen kysytään aina uuden käyttäjän saapuessa verkkopalveluun. Lupaa kysytään jokaisen sivustovierailun yhteydessä, mikäli käyttäjä ei nimenomaisesti anna lupaa paikantamiseen pysyvästi. Varsinaisesta käyttäjän paikantamisesta vastaa selainohjelmisto. Paikkatiedon lähteenä selain voi käyttää esimerkiksi GPS, WiFi, IP, GSM/CDMA tai Bluetooth -pohjaisia paikannusmenetelmiä. HTML5-geopaikannus rajapinnan spesifikaatio ei kuitenkaan ota

kantaa paikannusmenetelmän tekniseen toteutukseen. Spesifikaatio kuitenkin vaatii, että rajapinnan täytyy palauttaa tieto paikannuksen tarkkuudesta. (W3C 2014.)

Kuva 6. *Käyttäjän sijainnin ja paikannuksen tarkkuuden visualisointi.*

Hakukoneen hakutulokset järjestetään ensisijaisesti etäisyyden mukaiseen järjestykseen. Hakuparametrina käytettyä sijaintia läheisin hakutulos listataan ensimmäisenä.

Yhteisön ylläpitämille hakutuloksille määritetään numeerisesti ilmaistavissa oleva laatu. Laatu muuttuu hakutuloksen elinkaaren aikana. Yksittäisen hakutuloksen laatuun vaikuttavia tekijöitä ovat:

- hakutulokseen liittyvät arvostelut
- hakutuloksen luojan maine.

Hakutulokseen liittyvät arvostelut ovat käyttäjäyhteisön hakutulokselle antamia plus- ja miinusääniä. Kukin käyttäjä voi antaa yhden äänen per hakutulos. Käyttäjät voivat myöhemmin muuttaa mieltään ja antaa äänensä uudelleen. Hakutulosten arviointi äänestämällä antaa yhteisölle suoran mahdollisuuden määritellä yksittäisten hakutulosten laadun. Äänestäminen luo hakukoneeseen Baba & Kashiman (2013) kuvaaman kaksivaiheisen laadunvarmistusjärjestelmän (vertaa kuva 1).

Hakutuloksen luontivaiheessa hakutuloksen laadulle annetaan lähtöarvo, joka määräytyy hakutuloksen luojan maineen perusteella. Hyvämaineisten käyttäjien voidaan olettaa luovan hyvälaatuisia hakutuloksia. Tästä syystä hakutuloksen laatua kuvaava arvo voidaan asettaa jo lähtötasoltaan paremmaksi kuin huonomaineisen käyttäjän luoman hakutuloksen. Tämä menettely myös palkitsee hyvämaineisia yhteisön jäseniä ja luo kannustimen hyvän maineen tavoitteluun.

Laatua käytetään hakutulosten painottamiseen. Hyvälaatuisten hakutulosten karttamerkki näytetään isompana ja heikkolaatuisten hakutulosten karttamerkki pienempänä tai se voidaan piilottaa kokonaan. Heikompilaatuisten hakutulosten piilottaminen tulee kyseeseen erityisesti silloin, kun rajatulla hakualueella on suuri määrä hakutuloksia ja kaikkien hakutulosten visualisointi ei ole käytännöllistä. Tällöin käyttäjän täytyy rajata hakualuetta tarkemmin nähdäkseen kaikista heikkolaatuisimmatkin hakutulokset.

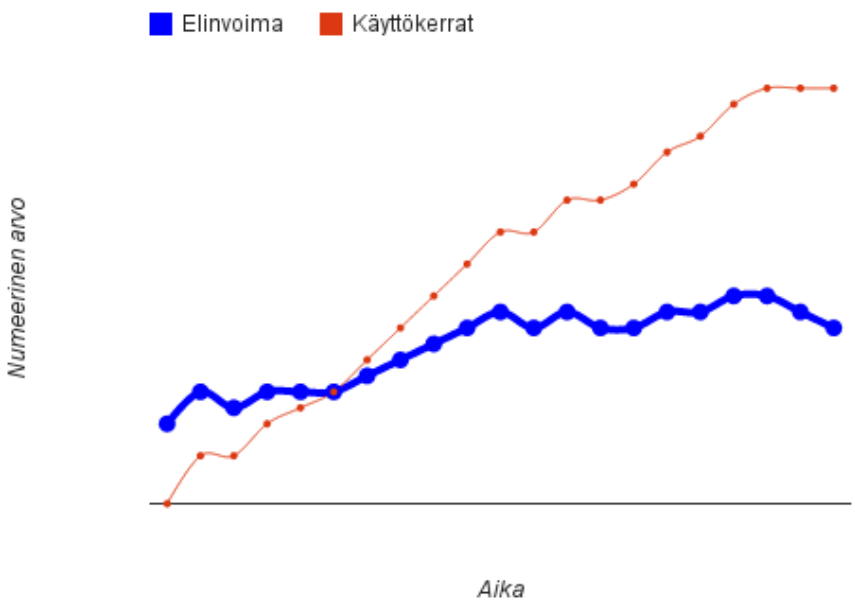
Hakutuloksen elinkaari alkaa hakutuloksen luonnista ja päättyy kun hakutulos poistetaan. Hakutulos poistetaan kun jompikumpi seuraavista ehdoista täyttyy: hakutuloksen laatua kuvaava numeerinen arvo putoaa negatiiviseksi tai hakutuloksen elinvoima loppuu.

Laatua kuvaava numeerinen arvo putoaa negatiiviseksi vain, jos käyttäjäyhteisön antamien äänten ja hakutuloksen laadun lähtöarvon summa on negatiivinen. Tällöin voidaan katsoa, että yhteisö on päättänyt hakutuloksen olevan niin huono, että se on syytä poistaa.

Hakutuloksella on numeerisesti ilmaistavissa oleva elinvoima E, joka vähenee ajan funktiona. Toisaalta jokainen hakutuloksen käyttökerta k lisää hakutuloksen elinvoimaa pidentäen hakutuloksen elinikää. Elinvoiman arvo ajanhetkellä t voidaan esittää muodossa:

$$E(t) = E_{t_0} + \sum_{i=t_0}^{t} k_i - a\Delta t, \tag{1}$$

jossa E_{t_0}on elinvoiman arvo ajanhetkellä $t_0$eli elinvoiman alkuarvo, k on hakutuloksen käyttökerrat päivän aikana, a on ikääntymisen painokerroin ja Δt hakutuloksen ikä päivinä. Muuttamalla ikääntymisen painokerrointa a hakutulosten poistumisnopeutta voidaan säätää.



Kuva 7. *Hakutuloksen elinkaarimalli.*

Kuvaajassa 7 havainnollistetaan esimerkin avulla hakutuloksen elinkaaren luonnetta. Kuvassa hakutuloksen kumulatiivisia käyttökertoja kuvaa ohuempi ja elinvoimaa kuvaa paksumpi käyrä. Esimerkki kuvaajasta nähdään, että hakutulokset vaativat jatkuvaa käyttöä pysyäkseen "hengissä". Ikääntymisen painokertoimella voidaan säätää esimerkiksi erittäin hyvälaatuiset hakutulokset "kuolemaan" hitaammin ja huonolaatuiset hakutulokset "kuolemaan" nopeammin.

Hakutuloksen käyttökerrat siis muodostavat hakutulokselle käyttöhistorian. Käyttöhistorialla tarkoitetaan sitä, kuinka usein käyttäjät ovat valinneet kyseisen hakutuloksen eli avanneet hakutuloksen sisältämän linkin ajan saatossa. Jokainen hakutuloslinkin aktivointi vaikuttaa hakutuloksen elinvoimaan. Eli hakutulokset vaativat jatkuvaa käyttöä säilyttääkseen elinvoimansa. Vähän käytetyt hakutulokset "kuolevat" automaattisesti pois kun niiden numeerinen elinvoima putoaa nollaan. Näin varmistetaan, että hakukone tarjoaa vain hakutuloksia joita hakujen tekijät myös käyttävät.

Hakutulosten käyttöhistoria ei voi olla hakutuloksen laadun osatekijä, koska korostamalla käyttäjien valitsemia hakutuloksia samalla myös lisättäisiin todennäköisyyttä, että hakutulos tulisi valituksi uudelleen myös muiden käyttäjien toimesta. Syntyisi itseään vahvistava kierre.

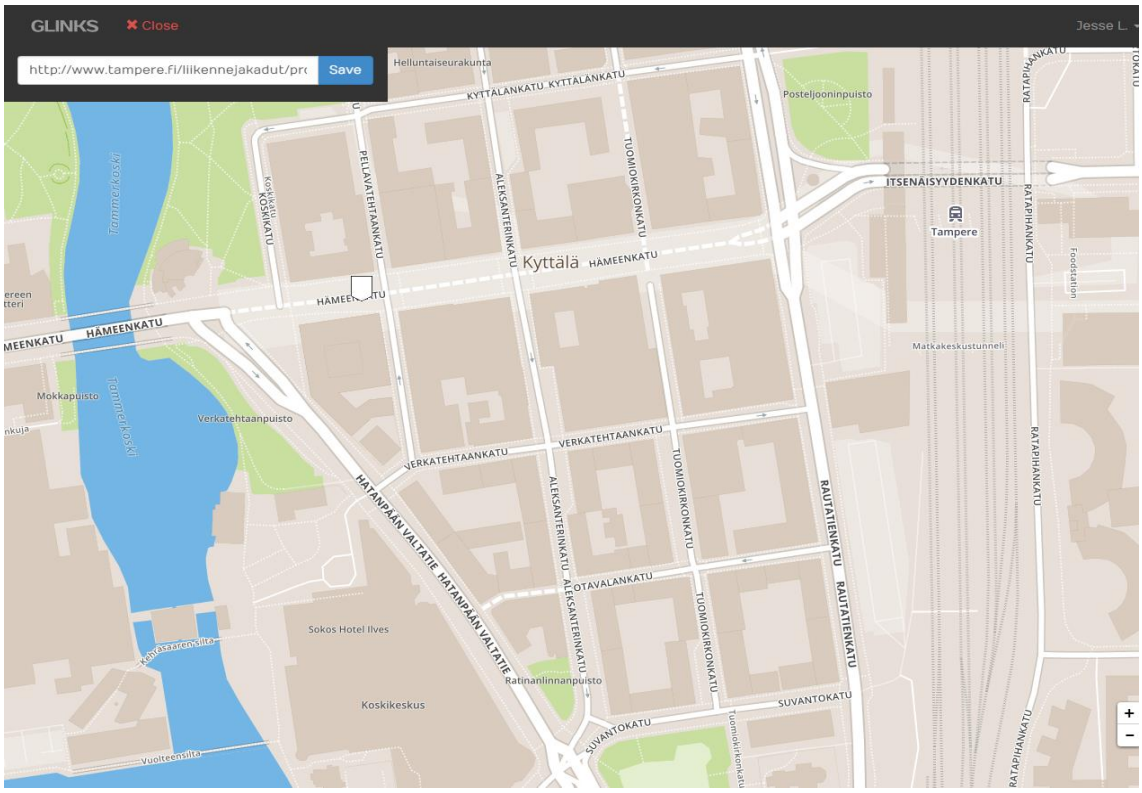
Käyttöhistoria antaa kuitenkin hyvän kuvan siitä kuinka relevantti hakutulos on ajallisesti. Tässä työssä suunnitellun hakukoneen tavoitteena on auttaa käyttäjiä löytämään ajantasaista tietoa Webistä ja tästä syystä vanhentuneet hakutulokset pyritään poistamaan hakutulosten elinkaarimallin avulla.

4.2 Prototyypin toiminnallinen kuvaus

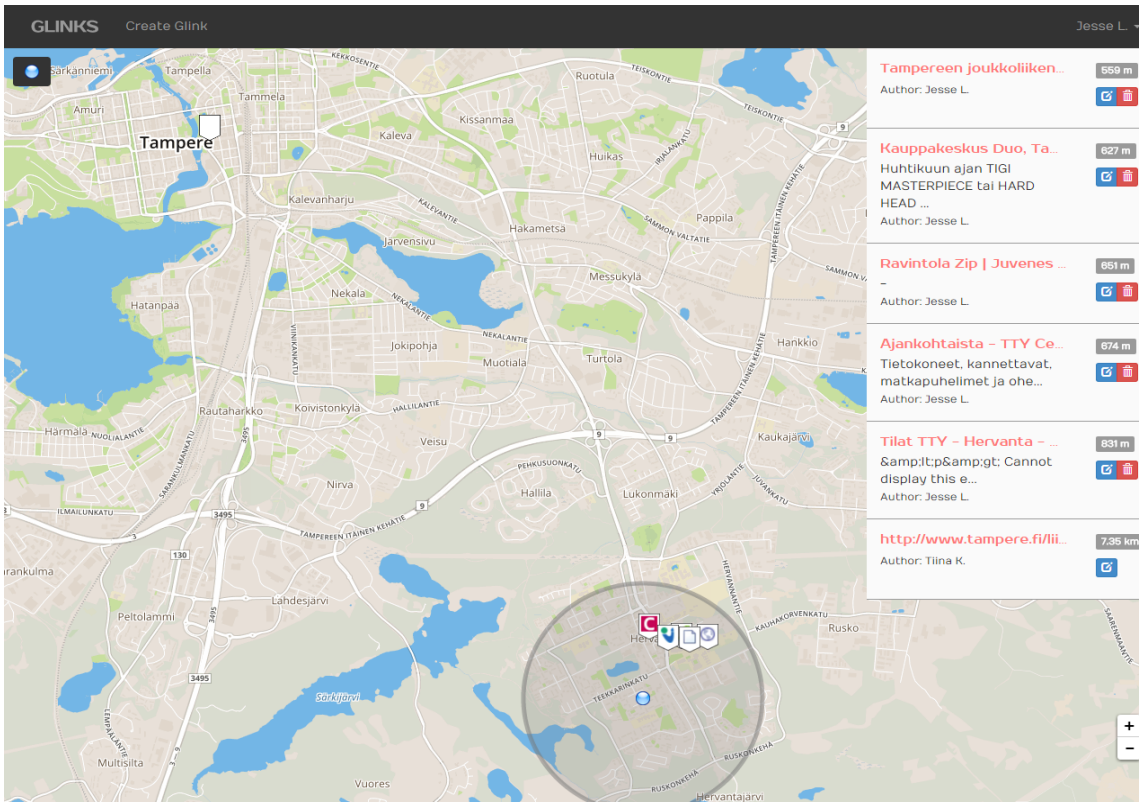
Tämän työn rinnalla toteutettu prototyyppi toteutettiin pienillä resursseilla ja tästä syystä siihen ei voitu ottaa mukaan kaikkia esimerkkitoteutuksen ominaisuuksia. Prototyyppiin toteutettiin kuitenkin perus ominaisuudet paikkatiedon keräämiseksi joukkoistamalla sekä perus ominaisuudet paikkapohjaisten hakujen tekemiseen.

Prototyyppi järjestelmään voi luoda käyttäjätilin. Käyttäjätilin luotuaan käyttäjä voi luoda karttakäyttöliitymän avulla geokoodattuja linkkejä. Kuvassa 8 on havainnollistettu geokoodatun linkin luominen. Käyttäjä valitsee kartalta pisteen, johon linkki lisätään ja syöttää linkin URL-osoitteen sille varattuun tekstikenttään vasemmassa ylälaidassa. Käyttäjän valitsema paikka merkitään karttamerkillä. Karttamerkkiä voi liikutella kartalla raahaamalla. Lopuksi käyttäjä painaa Save-painiketta, joka viimeistelee geokoodatun linkin luonnin. Käyttäjälle näytetään vielä viesti onnistuneesta geokoodatun linkin luonnista. Uutta geokoodattua linkkiä käytetään välittömästi hakutuloksena kaikille käyttäjille.

Vasta luodun hakutuloksen tiedoissa näytetään vain käyttäjän syöttämä URL-osoite ja hakutuloksen luojan nimi, koska Web-dokumenttien otsikko, lyhyt kuvaus liitetään hakutulokseen vastamyöhemmin. Tietoja ei siis haeta välittömästi geokoodatun linkin luonnin yhteydessä vaan ne haetaan Web-ryömijän avulla ajastetusti säännöllisin väliajoin. Käyttöliittymän kartalla näytetään myös käyttäjän oma sijainti ja arvio paikannuksen tarkkuudesta. Käyttäjä voi myös kohdistaa kartan omaan sijaintiinsa painamalla kuvan käyttöliittymästä löytyvää paikannuspainiketta. Painike on kuvattu kuvan 9 vasemmassa ylälaidassa.



Kuva 8. *Geokoodatun linkin luominen.*



Kuva 9. *Hakutulokset visualisoituna karttamerkeillä ja hakutuloslistana.*

Käyttäjälle näytetään hakutuloksia aina kartalle kohdistetun alueen mukaisesti. Tullessaan palveluun kartta kohdistetaan automaattisesti käyttäjän omaan sijaintiin, jos käyttäjä

sen sallii. Muussa tapauksessa kartta kohdistetaan Tampereelle. Käyttäjä voi kohdistaa kartan haluamaansa sijaintiin raahaamalla, osoittamalla ja zoomaamalla Webin karttakäyttöliittymissä totuttuun tapaan. Kartan hakutuloksia indikoivat karttamerkit ja hakutuloslistaus päivittyvät automaattisesti kartan rajaaman alueen muuttuessa.

Kuvassa 9 havainnollistetaan prototyypin karttakäyttöliittymää. Tiedonhaku Webistä tapahtuu yksistään karttaan kohdistetun alueen perusteella. Hakutulokset visualisoidaan favikoneilla rikastetuilla karttamerkeillä ja hakutuloslistana. Hakutuloslistan hakutuloksissa näytetään viitatun Web-dokumentin otsikko, lyhyt tekstikuvaus sisällöstä, hakutuloksen luojan nimi sekä etäisyys käyttäjän sijainnin ja hakutuloksen välillä. Lyhyet etäisyydet näytetään metreinä ja suuremmat etäisyydet kilometreinä.

Prototyypin käyttöliittymä on toteutettu responsiivista suunnittelua (Polacek 17.5.2015) noudattaen, jotta se skaalautuu hyvin erikokoisille näytöille. Paikallishaun pääkohderyhmä on mobiilikäyttäjät ja tästä syystä käyttöliittymän täytyy toimia myös mobiililaitteilla. Toteutuksessa on kuitenkin huomioitu vain kaikista nykyaikaisimmat mobiililaitteet.

Järjestelmään kirjautuneille käyttäjille näytetään hakutulosten yhteydessä myös muokkaa-painike, jonka avulla käyttäjä voi muokata hakutuloksen URL-osoitetta tai liikuttaa hakutuloksen karttamerkkiä. Lisäksi hakutuloksen luojalle näytetään muokkaa-painikkeen rinnalla poista-painike, jonka avulla käyttäjä voi poistaa koko hakutuloksen järjestelmästä.

5. TEKNINEN TOTEUTUS

Yhteisöllinen paikallishakukone on Webissä toimiva ohjelmistojärjestelmä, joka vaatii toimiakseen järjestelmän liiketoimintalogiikan toteuttavan palvelinohjelmiston sekä Internetin välityksellä toimivan käyttöliittymän. Tässä luvussa kuvataan näille paikallishakukoneen kriittisille osille tekninen esimerkkitoteutus. Tekninen toteutus kuvaa tämän työn ohella tehtyä paikallishakukoneen prototyyppiä.

5.1 Taustaa toteutustekniikoista

Ruby-ohjelmointikielelle[5] toteutettu Web-sovelluskehys Ruby on Rails[6] (Rails) on tunnettu lähinnä sen mainostetusta kehittäjäystävällisyydestä ja kehitysnopeudesta, mutta se on myös paljon käytetty ja arvostettu sovelluskehys. Railsillä on laaja ja aktiivinen käyttäjäyhteisö, joka kehittää alustaa jatkuvasti eteenpäin. Ruby-ohjelmointikielelle on tarjolla valtava määrä valmiita ohjelmistopakkauksia, jotka nopeuttavat Ruby-ohjelmien kirjoittamista huomattavasti. Näistä syistä Rails on oiva valinta yhteisöllisen hakukoneen Web-sovelluskehykseksi.

Yksittäistä ohjelmistopakkausta kutsutaan Ruby-yhteisössä termillä gem. Gemit ovat Rubyn uudelleenkäytettäviä avoimen lähdekoodin ohjelmistopakkauksia, joista suosituimmat ovat kattavasti testattuja, laajasti käytettyjä ja hyvin ylläpidettyjä. Toisaalta gemien joukossa on myös valtava määrä huonosti toteutettuja ja vailla ylläpitoa olevia pakkauksia. Gemien käyttäminen säästää aikaa uusien ominaisuuksien toteuttamisessa, kun kyseiset ominaisuudet saadaan käyttöön pienellä vaivalla valmiita pakkauksia hyödyntämällä tai ne voidaan koostaa hyödyntämällä muiden toteuttamia ominaisuuksia.

Testattujen ja ylläpidettyjen avoimen lähdekoodin ohjelmistopakkausten oletetaan usein myös johtavan parempaan laatuun, koska laaja ja aktiivinen yhteisö pitää huolen, että lähdekoodia ylläpidetään ja kehitetään jatkuvasti. Avoimen lähdekoodin parempi laatu suhteessa suljettuun lähdekoodiin on kuitenkin kiistanalainen aihe, josta ei ole laajaa tieteellistä näyttöä, kuten Raghunathan et al. (2005) toteavat työssään. Voidaan kuitenkin todeta, että avoimen lähdekoodin ohjelmistot kuten esimerkiksi Rails mahdollistaa laajojen ohjelmistojen tekemisen pienillä resursseilla. Avoin lähdekoodi mahdollistaa esimerkiksi sen, että tämän työn ohella toteutettu sosiaalisen hakukoneen prototyyppi Glinks pystyttiin toteuttamaan yhden miehen osaamisella. Ilman avointa lähdekoodia tai isoa budjettia tämä ei olisi mahdollista.

[5] https://www.ruby-lang.org/en/
[6] http://rubyonrails.org/

Ruby-gemien laatua voi arvioida lukemalla lähdekoodia. Koska lähdekoodin lukeminen ei ole aina kuitenkaan käytännöllistä, kehittäjät voivat arvioida ja etsiä gemejä muun muassa Ruby Toolbox -palvelusta[7], joka listaa gemeistä metatietoja kuten latausmääriä ja ylläpidon aktiivisuutta kuvaavia tietoja. Ruby Toolboxissa gemejä voi etsiä muun muassa kategorioiden avulla ja vertailla samoihin ongelmiin tarkoitettuja gemejä keskenään. Gemien lähdekoodi löytyy useimmiten GitHub-versiohallintapalvelusta. GitHub tarjoaa versionhallinnan lisäksi alustan gemien ylläpitoon liittyvälle keskustelulle, vikaraporteille ja työkaluja Open Source -kehitykselle.

5.2 Arkkitehtuurikuvaus

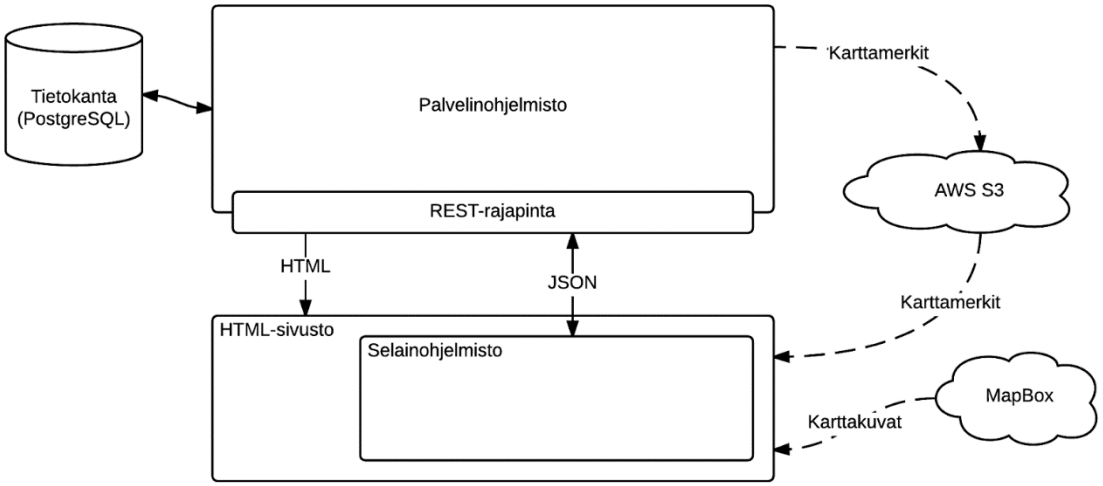
Sosiaalinen paikallishakukone pohjautuu Web-sovellusten mukaisesti asiakas-palvelinmalliin, jossa asiakkaana toimii Web-selain tai muu Internetin välityksellä palvelimelle keskusteleva ohjelmisto ja palvelimena Web-palvelin. Yksinkertaisuuden vuoksi tässä esimerkkitoteutuksessa oletetaan, että asiakkaana toimii Web-selain. Palvelimella toimivaa ohjelmistokokonaisuutta kutsutaan usein termeillä backend tai palvelinohjelmisto ja selaimessa toimivaa osuutta termeillä frontend tai selainohjelmisto. Alalla vakiintunut käytäntö on käyttää termien englanninkielisiä muotoja backend ja frontend, mutta tässä työssä pyritään käyttämään sanojen suomenkielisiä vastineita.

Hakukoneen liiketoimintalogiikka ja käyttöliittymä voidaan toteuttaa erillisinä kokonaisuuksina toteuttamalla palvelinohjelmisto REST-rajapintamallin mukaisella arkkitehtuurilla. Niin sanottu liiketoimintalogiikka tapahtuu tässä mallissa suurimmaksi osaksi palvelinohjelmistossa kun taas selainohjelmiston vastuulla on käyttöliittymän hallinta. Selainohjelmisto kommunikoi palvelinohjelmiston kanssa REST-rajapinnan välityksellä välittäen dataa JSON-formaatissa. Selainohjelmiston tärkein tehtävä on pitää hakukäyttöliittymässä näytettävä kartta ja hakutulokset ajan tasalla, välittää käyttäjän komennot palvelinohjelmistolle ja välittää paluuviestit käyttäjälle.

Selainohjelmisto hyödyntää kartan käsittelyssä Leaflet JavaScript -kirjastoa[8], joka on suunniteltu nimenomaisesti mobiiliystävällisten interaktiivisten karttojen toteuttamiseen. Leaflet tarjoaa rajapinnan karttanäkymän hallintaa. Leaflet ei itsessään ota kantaa, mitä karttapohjia karttanäkymässä halutaan hyödyntää vaan Leafletin tarjoama rajapinta tarjoaa keinot valita tarpeiden mukaisesti tietyn palveluntarjoajan karttapohjat.

Kuvassa 10 on kuvattu hakukoneen korkean tason arkkitehtuuri ja ulkoiset riippuvuudet. Komponenttien välistä tiedonkulkua on kuvattu nuoliviivoilla. Nuolen päät kuvaavat tiedon liikkumissuuntaa komponenttien välillä. Hakukoneen ulkoiset riippuvuudet on kuvattu katkoviivalla.

[7] https://www.ruby-toolbox.com/
[8] http://leafletjs.com/



Kuva 10. *Yhteisöllisen paikallishakukoneen keskeiset komponentit.*

Palvelinohjelmisto tallentaa hakutuloksille yksilöllisesti luodut karttamerkit Amazon Web Services (AWS) -pilvipalvelun Simple Storage Service (S3) -palveluun. Selainohjelmiston Leaflet JavaScript -kirjasto on yhteydessä MapBox-karttapalveluun, joka tarjoaa karttanäkymän karttapohjat. Palvelinohjelmiston sisäiset komponentit on kuvattu tarkemmin alakohdassa 5.2.1 ja selainohjelmiston komponentit alakohdassa 5.2.2.

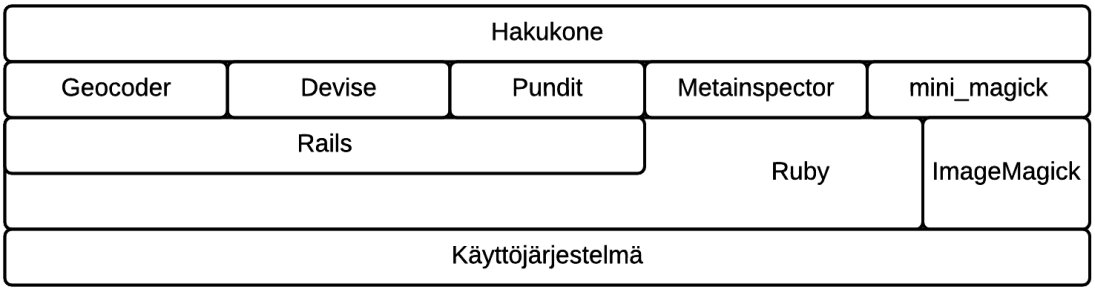
5.2.1 Palvelinohjelmisto

Hakukoneen palvelinohjelmisto koostuu Ruby on Railsin päälle toteutetusta sovelluksesta, joka on riippuvainen useista kolmansien osapuolien ohjelmistopakkauksista. Hakukoneen rakenne voidaan jakaa karkeasti hakutulosten ja käyttäjäyhteisön hallintaan. Hakutulosten hallinta pitää sisällään hakutulosten etsimisen, luomisen, muokkaamisen, arvioinnin ja poistamisen hallinnan. Tämän lisäksi selainohjelmistolle täytyy tarjota rajapinta, jonka avulla selainohjelmisto voi kysellä käyttäjän tekemiä hakuja vastaavat hakutulokset. Käyttäjäyhteisön hallinta sisältää käyttäjien maineen ja oikeuksien hallinnan sekä varsinaisten käyttäjätilien ylläpitämiseen liittyvät tehtävät.

Kuvassa 11 on kuvattu palvelinohjelmiston kerrosarkkitehtuuri. Kerrosarkkitehtuuri kuvaa hakukoneen rakentamiseen vaadittujen teknologioiden suhdetta toisiinsa. Kerrosarkkitehtuurissa ylempänä olevat teknologiat käyttävät alempien tasojen tarjoamia palveluita.

Geocoder-pakkaus tarjoaa kätevän rajapinnan tietomallien geokoodaamiseen eli koordinaattien määrittämiseen tekstuaalisen sijainnin perusteella, käänteiseen geokoodaamiseen eli tekstimuotoisen sijainnin määrittämiseen koordinaattien perusteella ja etäisyyteen perustuvien tietokantakyselyiden tekemiseen.

Devise on Railsin käyttäjien autentikointiin tarkoitettu de facto -ohjelmistopakkaus. Devise tarjoaa käyttäjätileihin liittyvät perusominaisuudet kuten, tilien luonnin, päivittämisen, sulkemisen, salasanan vaihtamisen käyttövalmiina.



Kuva 11. *Palvelinohjelmiston kerrosarkkitehtuuri.*

Pundit on käyttäjien oikeuksien hallintaan tarkoitettu ohjelmistopakkaus. Punditin ajatuksena on, että kullekin järjestelmän tietomallille voidaan määritellä erillinen käyttöoikeuspolitiikka.

```
class GlinkPolicy
  attr_reader :user, :glink

  def destroy?(user, glink)
    glink.author == user
  end

  def update?
   user == glink.author or user.reputation >= LIMIT_TO_UPDATE
  end
end
```

Ohjelma 1. *Esimerkki hakutuloksen käyttöoikeuspolitiikasta.*

Ohjelmassa 1 on havainnollistettu geokoodatun linkin eli `glinkin` käyttöoikeuspolitiikka. Politiikan mukaan `glinkin` voi tuhota käyttäjä, joka on `glinkin` luoja ja `glinkkiä` voi muokata `glinkin` luoja sekä käyttäjät joiden maine ylittää tietyn raja-arvon.

Metainspector-pakkaus tarjoaa korkean tason rajapinnan Web-dokumenttien raapimiseen ja dokumenttien metatietojen keräämiseen. Hakukone hakee metainspectorin avulla hakutulossivujen otsikon, lyhyen kuvaksen ja favikonin.

Mini_magick-pakkaus tarjoaa Ruby-rajapinnan *Imagemagick* kuvankäsittelyohjelmistokokonaisuuden käyttämiseen. Hakukone luo kaikille hakutuloksille, joille on saatavilla favikoni, yksilöllisen karttamerkin. Karttamerkki muodostetaan yhdistämällä karttamerkin pohjakuva ja sivuston favikoni. Tässä toimenpiteessä hyödynnetään Imagemagickin tarjoamia palveluita mini_magickin tarjoaman rajapinnan välityksellä.

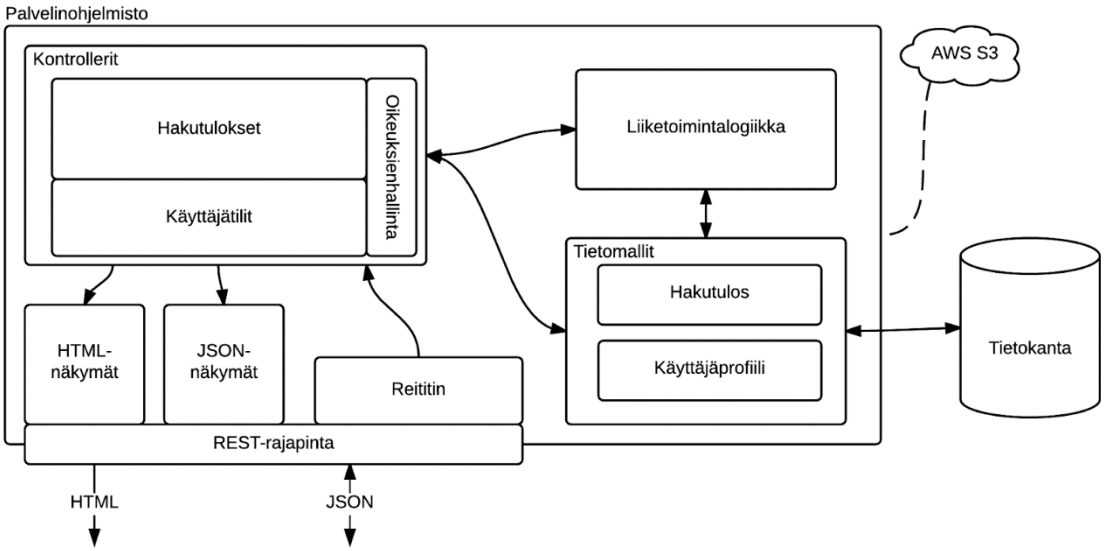
Ohjelmassa 2 on kuvattu favikonin ja karttamerkin pohjakuvan yhdistäminen mini_magick gemin avulla. Varsinaisen kuvankäsittely työn tekee `composite` funktio, joka yhdistää kuvat annettujen parametrien mukaisesti.

```
def generate_marker
    "Generate map marker icon from the base image and favicon"
    blank_marker_path = 'app/assets/images/blank_marker.png'
    blank_marker = MiniMagick::Image.open(blank_marker_path)

    fav_icon = MiniMagick::Image.open(glink.get_favicon_url())
    fav_icon.resize('48x48')

    blank_marker.composite(fav_icon) do |c|
      c.compose 'Over'
      c.geometry '+8+8'
    end
  end
```

Ohjelma 2. *Karttamerkin luonti.*

Rails tarjoaa hakukoneen vaatimat Web-sovellus ominaisuudet. Hakukoneen kannalta Railsin keskeisimpiä palveluita ovat ActiveRecord-mallin mukainen olio-relaatio-kartoitus (engl. object relational mapping) ja sovelluskehyksen tarjoama MVC-malli. ActiveRecord tarjoaa korkean tason rajapinnan Ruby-olioiden ja relaatiotietokannan taulujen hallintaan ja kyselyjen tekemiseen.

Kuvassa 12 on kuvattuna palvelinohjelmiston keskeiset komponentit ja niiden väliset suhteet. Komponenttien välistä kommunikointia on havainnollistettu nuoliviivoilla, joissa nuolenpää kuvaa tiedonsiirron suuntaa.



Kuva 12. *Palvelinohjelmiston korkean tason arkkitehtuuri.*

Hakukone pohjautuu Railsin MVC-arkkitehtuuriin (C2 17.5.2015). REST-rajapinnan (Fielding 2000) vastaanottamat HTTP(S)-pyynnöt ohjataan sovellukseen määritellyn reititys logiikan perusteella oikealle kontrollerille. Kullekin hakukoneen tietomallille on oma kontrollerinsa, joka keskustelee oikeuksien hallinnasta vastaavan komponentin kanssa varmistaakseen käyttäjän oikeudet pyydettyyn toimenpiteeseen. Kontrollerit ohjaavat sallitut toimenpidepyynnöt tietomallien ja liiketoimintaloogiikan toteutettavaksi. Tietomallit keskustelevat Railsin oliorelaatiorajapinnan avustuksella tietokannan kanssa. Toimenpiteiden HTTP(S)-paluuviestit muodostetaan palvelinohjelmistoon määriteltyjen HTML- ja JSON-näkymien avulla.

5.2.2 Selainohjelmisto

Hakukoneen käyttöliittymä toteutetaan pääosin selainohjelmistossa. JavaScriptillä toteutetun käyttöliittymän avulla, säästytään ylimääräisiltä sivunlatauksilta, mikä madaltaa käyttöliittymän vasteaikoja ja tekee käyttökokemuksesta mielekkäämmän. Näin saadaan myös vähennettyä siirrettävän datan määrää palvelimen ja selaimen välillä, koska palvelimelta ei tarvitse ladata uusia HTML-dokumentteja poislukien ensimmäinen sivunlataus. Riittää, että palvelimelta ladataan vain tapahtumien vaatima JSON-muotoinen data. Nopeampi vasteaika antaa käyttäjälle paremman käyttökokemuksen ja vähäisempi tietoliikenteen määrä on erityisen tavoiteltavaa etenkin mobiilikäyttäjien kohdalla. Siirtämällä ominaisuuksia selainohjelmiston vastuulle palvelimeen kohdistuvan laskentatehon tarve pienenee vastaavasti, jolloin palvelinohjelmisto voi palvella useampia käyttäjiä pienemmillä resursseilla.

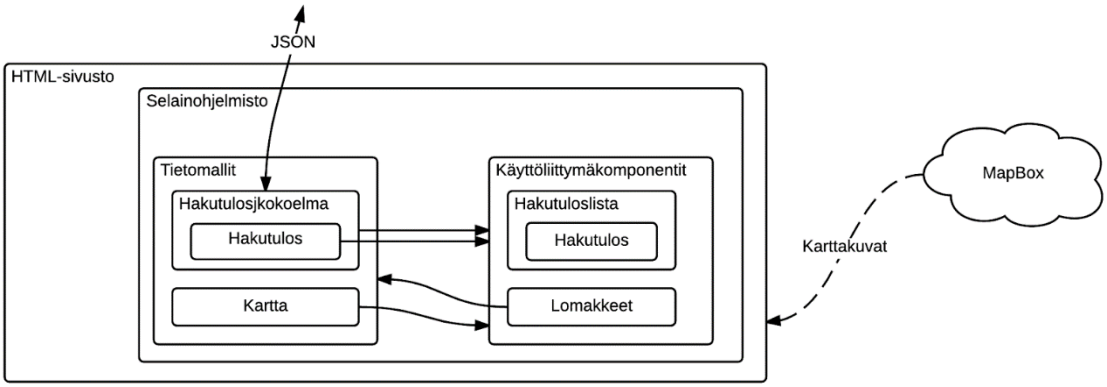
Käyttöliittymä logiikan siirtäminen selaimeen sisältää myös haasteita. Selainohjelmistoon joudutaan toteuttamaan osittain päällekkäisiä toimintoja palvelinohjelmistojen kanssa. Esimerkiksi käyttäjän syötteiden tarkastus on syytä tehdä sekä selain- että palvelinohjelmistossa. Palvelinohjelmistossa toteutetut tietomallit täytyy mallintaa uudelleen selainohjelmistossa, jotta niistä luotujen olioiden hallitseminen on käytännöllistä. Tietomallien tapauksessa palvelinohjelmiston lähettämä JSON-muotoon serialisoitu Ruby-olio uudelleen instantioidaan JavaScript-olioksi selainohjelmistossa. JSON-serialisoinnin yhteydessä olion Rubyllä toteutettuja metodeja ei pystytä kätevästi siirtämään selainohjelmistoon.

Kuva 13 havainnollistaa selainohjelmiston arkkitehtuuria. Kuvauksesta käy ilmi selainohjelmiston keskeisimmät osat ja niiden väliset suhteet. Nuoliviivat kuvaavat komponenttien välisen tiedonsiirron suuntaa.

Selainohjelmisto on sisällytettynä hakukoneen HTML-sivustossa. Puhtaan HTML-sivuston avulla voidaan toteuttaa hakukoneen perusominaisuuksia, joille ei ole erityistä tar-

vetta JavaScript toteutukselle, mutta tämän työn ohella tehdyssä prototyyppi hakukoneessa vaaditaan JavaScript tukea. Selainohjelmisto sisältää ohuen pääohjelman, joka alustaa selainohjelmistossa käytetyt tietomallit ja käyttöliittymän.

Hakutuloskokoelma on selainohjelmiston keskeisin tietomalli, joka tarjoaa rajapinnan hakutulosjoukon hallintaan. Kokoelma sisältää siis joukon yksittäisiä *hakutuloksia*, jotka edustavat kulloinkin voimassa olevan haun hakutuloksia. Hakuparametrien muuttuessa hakutuloskokoelma päivitetään palvelimelta saadun vastauksen mukaisesti. Keskustelu palvelinohjelmiston ja selainohjelmiston välillä tapahtuu asynkronisesti JSON-formaatin muodossa. Asynkroniseen viestintään hyödynnetään jQueryn tarjoamaa AJAX-rajapintaa.



Kuva 13. *Selainohjelmiston arkkitehtuuri.*

Kartta-tietomalli rakentaa Leaflet-karttakirjaston päälle hakukoneen ominaisuuksia varten korkeamman tason rajapinnan, jonka avulla kartta-tietomalliin voidaan muun muassa lisätä ja poistaa hakutuloksia, visualisoida käyttäjän sijainti ja tukea uusien hakutulosten luontia. Kartta-tietomalli lähettää myös signaaleja oman tilansa muutoksista, joihin muun muassa hakutuloskokoelma reagoi hakemalla uusia hakuparametreja vastaavat hakutulokset.

Käyttöliittymäkomponentit pohjautuvat Facebookin kehittämään React[9] JavaScript-käyttöliittymäkirjastoon. React tarjoaa rajapinnan, jonka avulla voidaan määritellä deklaratiivisesti käyttöliittymäkomponentteja. Käyttöliittymä muodostetaan joukosta pieniä uudelleen käytettäviä komponentteja. Kullakin komponentilla on sisäinen tila. Tilan muuttuessa React huolehtii automaattisesti, että käyttöliittymään piirretty komponentti päivitetään. Komponentit muodostavat hierarkian, jossa ylemmän tason komponentit määrittävät hierarkiassa alempana olevien komponenttien ominaisuudet. Ylemmän tason komponentin tilan muuttuessa, React huolehtii, että kaikki alemman tason komponentit päivittyvät myös.

[9] https://facebook.github.io/react/

Käyttöliittymäkomponentit kuuntelevat tietomallien lähettämiä signaaleja ja muuttavat tilaansa signaalien mukaisesti. Esimerkiksi hakutuloskokoelman saadessa uuden joukon hakutuloksia palvelinohjelmistolta, se lähettää signaalin *updated*. Karttakomponentti kuulee hakutuloskokoelman lähettämän *updated* signaalin ja päivittää oman tilansa. Koska hakutuloslistakomponentti on komponenttihierarkiassa karttakomponentin alla, myös hakutuloslista päivittyy automaattisesti *updated* signaalin seurauksena.

Käyttöliittymäkomponenttien joukossa on myös *lomakkeita*. Lomakkeet ohjaavat käyttäjien syötteet tietomalleihin. Tietomallit kommunikoivat tarvittaessa palvelinohjelmistolle käyttäjän toimenpiteistä. Tietomallien muuttuessa käyttöliittymäkomponentit päivittyvät saamiensa signaalien perusteella. Esimerkiksi uutta hakutulosta luodessa käyttäjä syöttää hakutuloksen tiedot hakutuloslomakkeeseen. Lomake ohjaa syötteet suoraan uutta hakutulosta edustavaan olioon. Lomakkeen ollessa valmis hakutulos olio yritetään tallentaa. Onnistuneen tallennuksen seurauksena olio lähettää signaalin, jota hakutuloskokoelma kuuntelee.

Käyttöliittymän kartta rakentuu pitkälti Vladimir Agafonkin kehittämän Leaflet-karttakirjaston päälle. Leaflet on Web-karttojen toteuttamiseen suunniteltu mobiiliystävällinen avoimen lähdekoodin JavaScript-kirjasto. Leaflet tarjoaa rajapinnan, jonka avulla karttakäyttöliittymän perustoiminnallisuudet, kuten kartan piirtäminen, liikuttaminen ja zoomaus on yksinkertaista toteuttaa. Rajapinnan avulla voidaan myös hallita karttaa ohjelmallisesti, lisätä objekteja kartalle ja reagoida käyttäjän toimenpiteisiin kartalla.

6. ARVIOINTI

Glinks-prototyyppi joukkoistamiseen pohjautuvasta paikallishakukoneesta ehti olla 9 päivää käytettävissä tämän työn puitteissa. Kuten Bruns & Bahnisch (2009) ohjeisti, hakukoneeseen lisättiin esimerkkisisältöä ohjaamaan ensimmäisten käyttäjien toimintaa. Esimerkki sisältönä luotiin 10 geokoodattua linkkiä, jotka antoivat ensimmäisille käyttäjille käsityksen siitä miltä hakutulokset näyttävät ja minkä tyyppistä sisältöä hakukoneeseen on hyvä lisätä.

Yhdeksän päivän aikana järjestelmään kirjautui 8 käyttäjää, jotka loivat yhteensä 52 hakutuloksena käytettävää geokoodattua linkkiä esimerkkien lisäksi. Taulukossa 1 on listattuna luotujen hakutulosten määrät per käyttäjä. Käyttäjien tunnisteena on käytetty järjestelmän automaattisesti luomaa tunnistetta käyttäjien anonymiteetin turvaamiseksi.

Taulukko 1. *Luotujen hakutulosten määrä per käyttäjä.*

Käyttäjän tunniste	0	1	2	3	4	5	6	7
Luodut hakutulokset (kpl)	14	6	4	9	5	3	9	2

Taulukosta 1 voidaan nähdä, että kaikki järjestelmään kirjautuneet käyttäjät loivat hakutuloksia. On kuitenkin huomioitava, että käyttäjät olivat ennestään tuttuja ja heitä oli erikseen pyydetty luomaan sisältöä hakukoneeseen tätä työtä varten.

Luoduista hakutuloksista 2 linkittivät uutisartikkeliin, 25 yritysten kotisivuille, 2 Youtube-videoon, 3 Facebook-sivulle, 1 Wikipediaan ja loput 19 hakutulosta linkittivät valokuviin, erilaisten yhteisöjen sivuille ja muihin vaikeasti luokiteltaviin sivuihin. Hakutuloksista löytyy myös kaksi duplikaattia. Käyttäjien luomien hakutulosten URL-osoitteet on listattuna kokonaisuudessaan liitteessä A.

Mielenkiintoista oli se, että luodut hakutulokset liittyivät käyttäjien omaan lähiympäristöön. Esimerkiksi Helsingissä asuva ja Porvoossa työskentelevä käyttäjä loi hakutuloksia Helsinkiin ja Porvooseen. Turussa asuva ja Tukholmassa vieraillut käyttäjä loi puolestaan hakutuloksia Turkuun ja Tukholmaan. Pääsääntöisesti kaikki hakutulokset lisättiin myös maantieteellisesti relevantteihin sijainteihin eli yhteys Web-dokumenttien ja sijaintien välillä oli selkeästi ymmärrettävissä. Yksi hakutuloksista oli selkeästi muista poikkeava, sillä siinä linkitettiin nyrkkitappelua kuvastava kuva Porin keskustaan. Nyrkkitappelu ja Porin maine yhdistetään kyllä usein toisiinsa, mutta yhteys kyseisen kuvan ja maantieteellisen sijainnin väillä vaati jo syvällisempää populaarikulttuurin tuntemista. Voidaan

todeta, että ainakin 51 hakutulosta 52:sta oli varsin laadukkaita. Ne yhdistivät selkeällä tavalla Web-dokumentin ja maantieteellisen sijainnin toisiinsa.

Teknisesti prototyypin toteutus oli toimiva. Arkkitehtuuri ratkaisut tukivat hyvin karttakäyttöliittymän vaatimuksia ja hakukone ominaisuuksien käyttäminen ei uusia sivunlatauksia. Prototyypin avulla pystyttiin testaamaan joukkoistamiseen pohjautuvan paikallishakukoneen perustoimintoja käytännössä. Hakutulosten lisääntyessä myös puuttuvien ominaisuuksien tarve näkyi selkeämmin. Hakutulosten kasautuessa pienelle alueelle niitä oli vaikea erottaa toisistaan. Hakutuloksia pitäisi pystyä piilottamaan ja painottamaan niiden laadun perusteella ja visualisoimaan selkeämmin. Käyttäjille pitäisi myös antaa palautetta heidän osallistumisestaan, jotta käyttäjät näkisivät selkeämmin oman aktiivisuutensa vaikutukset.

7. YHTEENVETO

Viime vuosikymmenien aikana Webin tietomäärä on kasvanut valtavasti. Tietoa on siis valtavasti saatavilla, mutta tieto on suhteellisen hyödytöntä, jos sitä ei ole helppo löytää. Tavoitteena on, että käyttäjää kiinnostavan tiedon löytäminen onnistuu mahdollisimman nopeasti, mahdollisimman pienellä vaivalla ja luontevalla tavalla. Webin yleiskäyttöiset avainsanahakukoneet mahdollistavat tämän useissa tapauksissa. Mobiililaitteiden yleistyessä käyttäjät etsivät tietoa kuitenkin yhä useammin omasta ympäristöstään. Avainsanahakukoneet eivät ymmärrä hyvin Web-dokumentteihin liittyvää paikkatietoa ja tarjoavat tästä syystä epärelevantteja hakutuloksia. Webin käyttäjillä ei ole tällä hetkellä tehokasta tapaa etsiä omaan ympäristöönsä tai muuhun maantieteelliseen sijaintiin liittyvää tietoa Webistä.

Useat hakukoneet kyllä tukevat paikallisten hakujen tekemistä, mutta niiden laajuus on pääsääntöisesti varsin rajoitettu toimialan tai tietotyypin mukaan. Toimivia paikallishakukoneita löytyy esimerkiksi yritysten, Wikipedia artikkeleiden, twiittien, Instagram-kuvien ja muiden sosiaalisen median viestien etsimiseen. Webin monimuotoisen tiedon etsiminen paikkapohjaisesti on kuitenkin edelleen pitkälti ratkaisematon ongelma. Ongelma on saanut useat yritykset ja tutkijat viimeisen reilun vuosikymmenen aikana liikkeelle, koska ongelman ratkaisemisella on suuria taloudellisia ja tieteellisiä kannustimia.

Yritysten toiminta ongelman ympärillä on pitkälti arvailujen varassa eikä tarkkaa tietoa yritysten kehittämistä algoritmeista ja tekniikoista ole saatavilla. Kuten myös Ahlers (2012) toteaa, saatavilla oleva tieto liittyy lähes pääsääntöisesti hakukoneiden käyttäjille näkyviin ominaisuuksiin eikä taustalla toimivista järjestelmistä kerrota juuri mitään. Tästä syystä kaupallisten hakukoneiden datalähteitä ja datan keräys- ja käsittelyalgoritmeja voi vain yrittää päätellä hakukoneiden julkisesti näkyvästä toiminnasta.

Tieteen puolella tutkijat ovat kehittäneet erilaisia keinoja kerätä dataa paikallishakukoneen tueksi, joista lupaavin lähestymistapa näyttäisi olevan paikkatiedon eristäminen Web-dokumenteista erilaisten säännöllisten lausekkeiden avulla. Vaikka eksplisiittisesti määritettyä paikkatietoa ei Webistä juurikaan löydy, niin tutkijat ovat onnistuneet keräämään muun muassa Web-dokumenttien sisältämiä osoitetietoja varsin tehokkaasti. Osoitetietojen avulla dokumentit voidaan liittää varsin tarkasti tiettyyn maantieteelliseen sijaintiin. Tässäkin lähestymistavassa on omat haasteensa. Esimerkiksi osoitetietojen eristäminen vaatii usein maakohtaista osoitetuntemusta ja tästä huolimatta osa osoitetiedoista jää löytämättä tai on epäluotettavaa (Ahlers & Boll 2008, Vänskä 2004). Lisäksi osoitetietoja löytyy usein vain tietyltä sivuston sivulta, vaikka koko sivusto liittyisi tiettyyn sijaintiin. Tämä lähestymistapa ei myöskään toimi alueilla, joissa osoitejärjestelmä puut-

tuu. Esimerkiksi suurin osan maailman meri ja maa-alueista jää perinteisen osoitejärjestelmän ulkopuolelle. Myös näihin alueisiin liittyvää tietoa löytyy Webistä. Parhaimmillaankin osoitteiden eristämisellä saadaan vain pieni osa Webin paikkatiedosta käyttöön.

Nykyiset paikallishakukoneet käyttävät hyödykseen myös paikkatietorekistereitä. Paikkatietorekisterit eivät sisällä lähtökohtaisesti kuitenkaan viittauksia Webiin. Tästä syystä paikkatietorekistereistä on korkeintaan tukea muille tekniikoille Web-paikkatietohakukonetta toteutettaessa. Backstrom et al. (2008) esitti kuinka hakukoneiden lokitietoja voidaan hyödyntää esimerkiksi hakutulosten painotuksessa. Hakulokien analysointi tarjoaa vain suuntaa antavaa tietoa hakuihin liittyen, eikä tieto ole tarpeeksi yksiselitteistä Web-paikallishakukoneen toteuttamiseen. Tekniikasta voi kuitenkin olla merkittävää hyötyä hakutulosten painotuksessa.

Prototyyppi yhteisöllisestä joukkoistamiseen pohjautuvasta paikallishakukoneesta ehti olla tämän työn puitteissa käytettävissä vain 9 päivää. Se on liian lyhyt aika, jotta voitaisiin tehdä mitään merkittäviä johtopäätöksiä ratkaisun toimivuudesta. Voidaan kuitenkin todeta, että ennalta tunnetut ja hyväntahtoiset käyttäjät loivat järjestelmään varsin laadukkaita hakutuloksia. Voidaan myös todeta, että järjestelmän alkuvaiheessa käyttäjillä on varsin vähän motivaatiota etsiä järjestelmästä uutta tietoa, koska hakutuloksia on vielä varsin vähän. Kyseessä on eräänlainen muna-kana-ongelma. Ensin täytyy saada sisältöä, jotta järjestelmään saadaan lisää käyttäjiä, mutta toisaalta käyttäjiä tarvitaan, jotta järjestelmään saadaan sisältöä. Vaaditaan niin sanottu lumipalloefekti yhteisön liikkeelle saattamiseen.

Järjestelmä toimii jo alkuvaiheessa hyvin omien kirjanmerkkien tallentamiseen tiettyyn sijaintiin. Kenties tällä käyttötarkoituksella voidaan motivoida ensimmäisiä käyttäjiä osallistumaan ja luomaan hakutuloksia. Myöhemmin järjestelmän fokus voidaan siirtää omien kirjanmerkkien hallinnasta lähemmäksi yleiskäyttöistä Web-paikallishakukonetta.

Tarkkoja tietoja ei ole julkisesti saatavilla, mutta on arvioitu, että Googlen indeksi sisältää noin 47 miljardia Web-sivua (WordlWideWebSize 17.2.2015). Yhtä ison indeksin kerryttäminen joukkoistamalla vaatisi valtavan määrän aktiivisia osallistujia ja jokaisen osallistujan pitäisi luoda valtava määrä linkkejä. Tämä ei liene kovin todennäköistä, mutta todennäköisesti jo huomattavasti pienempi määrä Web-sivuja listaava paikallishakukone tarjoaisi merkittävästi hyötyä, koska jokainen joukkoistamalla tuotettu hakutulos on jonkun käyttäjän näkökulmasta hyödyllinen. Googlen indeksi todennäköisesti puolestaan sisältää paljon suhteellisen vähäarvoisia sivuja.

On selkeää, että tehokas Web-paikallishakukone tarjoaisi täysin uudentyyppisen tavan etsiä tietoa ja helpottaisi paikallista tietoa vaativien sovellusten kehittämistä. Vielä sellaista ei ole pystytty toteuttamaan, mutta sen tuomat hyödyt ovat niin merkittäviä, että tutkimusta yleiskäyttöisen Web-paikallishakukoneen ympärillä on tärkeää jatkaa. Tässä

työssä esiteltyä ratkaisua ei ehditty tutkimaan tarpeeksi, jotta olisi voitu todeta sen toimivuus. Toisaalta kilpailevat ratkaisuvaihtoehdot eivät näytä tällä hetkellä tarjoavan laajamittaista ja tehokasta tukea yleiskäyttöiselle Web-paikallishakukoneelle. Tästä syystä ehdotan, että yhteisöllistä joukkoistamiseen pohjautuvaa Web-paikallishakukonetta tutkitaan pidemmällä aikavälillä tässä työssä kuvatun esimerkkitoteutuksen mukaisella järjestelmällä.

LÄHTEET

Abdul-Rahman, A. & Hailes, S. (2000). Supporting Trust in Virtual Communities. System Sciences, Proceedings of the 33rd Annual Hawaii International Conference, 4–7 Jan. 2000, Island of Maui. IEEE. pp. 1–9.

Ahlers D. & Boll S. (2007). Location-based Web Search. In: Prof. Scharl A. & Prof. Tochtermann K. The Geospatial Web - Advanced Information and Knowledge Processing. Springer London. pp. 55–66.

Ahlers D. & Boll S. (2008). Retrieving Address-based Locations from the Web. GIR '08 Proceedings of the 2nd international workshop on Geographic information retrieval, Napa Valley, California, USA, October 29–30, 2008. pp. 27–34.

Ahlers D. (2012). Chapter 3 Local Web Search Examined. In: Dirk Lewandowski (ed.), Web Search Engine Research. Library and Information Science, Volume 4. Emerald Group Publishing Limited. pp. 47–78.

Alnemr, R. & Meinel, C. (2011). Why rating is not enough: A study on online reputation systems. Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 7th International Conference, Orlando, FL, 15-18 Oct. 2011. pp. 415–421.

Baba, Y. & Kashima, H. (2013). Statistical Quality Estimation for General Crowdsourcing Tasks. KDD '13 Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. Chicago, Illinois, USA, August 11-14. ACM New York, NY, USA .2013. pp. 554–562.

Backstrom L., Kleinberg J., Kumar R. & Novak J. (2008). Spatial Variation in Search Engine Queries. Proceedings of the 17th international conference on World Wide Web, April 21–25, 2008, Beijing, China. pp. 357–366.

Bishob, J.(2013). The Psychology of Trolling and Lurking: The Role of Defriending and Gamification for Increasing Participation in Online Communities Using Seductive Narratives. In: J. Bishop (Ed.) Examining the Concepts, Issues, and Implications of Internet Trolling. IGI Global.

Brin, S., Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine, Computer Networks and ISDN Systems 30, pp. 107–117.

C2. Model View Controller [WWW]. [Viitattu: 17.5.2015]. Saatavissa: http://c2.com/cgi/wiki?ModelViewController

Chi, E. H. & Bernstein, M. S. (2012). Leveraging Online Populations for Crowdsourcing. Internet Computing, IEEE, Volume: 16, Issue: 5. pp. 10–12.

Cho, D., Kim, S. (2012). Empirical Analysis of Online Anonymity and User Behaviors: the Impact of Real Name Policy. 45th Hawaii International Conference on System Sciences. pp. 3041–3050.

Daviel A. & Kaegi F. (2007). Geographic registration of HTML documents [WWW]. Internet draft (IETF). Vancouver Webpages. [Viitattu: 7.5.2015]. Saatavissa: http://tools.ietf.org/id/draft-daviel-html-geo-tag-08.txt

Dean B. 2015. Google's 200 Ranking Factors: The Complete List [WWW]. [Viitattu: 12.5.2015]. Saatavissa: http://backlinko.com/google-ranking-factors

Dischler J. (2015). Building for the next moment [WWW]. Google. [Viitattu: 8.5.2015]. Saatavissa: http://adwords.blogspot.fi/2015/05/building-for-next-moment.html

Dlugolinsky S., Laclavik M. & Hluchy L. (2010). Towards a search system for the Web exploiting spatial data of a web document. Workshops on Database and Expert Systems Applications (DEXA), Bilbao, Aug. 30 2010-Sept. 3 2010. IEEE. pp. 27–31.

Dr Bruns, A. & Bahnisch, M. (2009). Smart services crc, Social Media: Tools for User-Generated Content, Social Drivers behind Growing Consumer Participation in User-led Content Generation, Volume 1 -State of the Art, March 2009. 60 p.

DuckDuckGo direct queries per day [WWW]. [Viitattu: 4.5.2015]. Saatavissa: https://duckduckgo.com/traffic.html

Dunn, J. (1984). The Concept of Trust in the Politics of John Locke. In: R. Rorty, J. B. Schneewind and Q. Skinner (eds.), Philosophy in History. Cambridge University Press, Cambridge, 1984. pp. 279–302.

Facebook. (2015). Statement of Rights and Responsibilities [WWW]. [viitattu 23.4.2015]. Saatavissa: https://www.facebook.com/legal/terms

Fielding, R. T. (2000). Architectural Styles and the Design of Network-based Software Architectures. Doctoral Dissertation. University of California, Irvine. pp. 1–162.

Fränti P., Tabarcea A., Kuittinen J. & Hautamäki V. (2010). Location-based Search Engine for Multimedia Phones Multimedia and Expo (ICME), 2010 IEEE International Conference, Suntec City, 19-23 July 2010. pp. 558–563.

Google Support. Verify a local business on Google [WWW]. [Viitattu 12.5.2015]. Saatavissa: https://support.google.com/business/answer/2911778?hl=en

Hosseini, M., Phalp, K., Taylor, J. & Ali, R. (2014). The Four Pillars of Crowdsourcing: a Reference Model. IEEE Eighth International Conference on Research Challenges in Information Science (RCIS), Marrakech, 28-30 May, 2014. IEEE. pp. 1–12.

IETF. Hostory for draft-daviel-html-geo-tag-08 [WWW]. [Viitattu: 12.5.2015]. Saatavissa: https://datatracker.ietf.org/doc/draft-daviel-html-geo-tag/history/

Iriberri, A. & Leroy, G. (2009). A Life-Cycle Perspective on Online Community Success. ACM Computing Surveys, Vol. 41, No. 2, Article 11, Publication date: February 2009, ACM New York, NY, USA. 29 p.

Khan J. A., Sangroha D., Ahmad M. & Rahman Md.T. (2014). A Performance Evaluation of Semantic based Search Engines and Keyword based Search Engines. Medical Imaging, m-Health and Emerging Communication Systems (MedCom), 2014 International Conference, Greater Noida, 7-8 Nov. 2014. IEEE. pp. 168–173.

Killoran J. B. (2013). How to Use Search Engine Optimization Techniques to Increase Website Visibility. Professional Communication, IEEE Transactions on Volume: 56, Issue: 1. pp. 50–66.

Kollock, P. & Smith, P. (1996). Managing the virtual commons: Cooperation and conflict in computer communities. In: S. Herring (Ed.), Computer-mediated communication: Linguistic, social, and crosscultural perspectives, Amsterdam: John Benjamins. pp. 109–128.

Lai L-F., Wu C-C., Lin P-Y. & Huang L-T. (2011). Developing a fuzzy search engine based on fuzzy ontology and semantic search. Fuzzy Systems (FUZZ), 2011 IEEE International Conference, Taipei, 27-30 June 2011. IEEE. pp. 2684–2689.

Lampe, C., Wash, R., Velasquez, A. & Ozkaya, E. (2010). Motivations to Participate in Online Communities, Michigan State University, College of Cummunication Arts and Sciences, April 10–15, 2010, Atlanta, Georgia, USA. ACM. pp. 1927–1936.

Lee, Y.W. (2003). Crafting Rules: Context-Reflective Data Quality Problem Solving, Journal of Management Information Systems, 20(3), pp. 93–119.

Loglisci C., Ienco D., Roche M., Teisseire M. & Malerba D. (2012). Toward Geographic Information Harvesting: Extraction of Spatial Relational Facts from Web Documents. Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference, Brussels, 10-10 Dec. 2012. IEEE. pp. 789–796.

Ludford, P. J., Cosley, D., Frankowski, D., Terveen, L. (2004). Think Different: Increasing Online Community Participation Using Uniqueness and Group Dissimilarity. CHI '04 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 631–638.

Lukyanenko, R. (2012). Crowd IQ: An Information Modeling Approach to Increasing Quality of User-generated Content. Thesis Proposal. Faculty of Business Administration. Memorial University of Newfoundland. 48 p.

Mihm D. A Closer Look at the Local Search Data Providers [WWW]. MOZ. [Viitattu: 9.5.2015]. Saatavissa: https://moz.com/learn/local/local-search-data-providers

Netmarketshare. Desktop/Mobile Search Engine Market Share [WWW]. [Viitattu: 4.5.2015]. Saatavissa: https://www.netmarketshare.com/

Nonnecke, B., Andrews, D. & Preece, J. (2006). Non-public and public online community participation: Needs, attitudes and behavior, Electron Commerce Res 6. pp. 7–20.

O'Keefe, P. (2011). Usernames vs. Real Names on Your Community: Pros and Cons [WWW]. [viitattu 23.4.2015]. Saatavissa: http://www.managingcommunities.com/2011/01/20/usernames-vs-real-names-on-your-community-pros-and-cons/

Polacek, J. What The Heck Is Responsive Web Design? [WWW]. [Viitattu: 17.5.2015]. Saatavissa: http://johnpolacek.github.io/scrolldeck.js/decks/responsive/

Quazilbash, N.Z., Qadri, S.M.H. & Khoja, S. (2012). Improved user RTSE experience on the web through fast retrieval of social media content. 15th International Multitopic Conference (INMIC). 13–15 Dec. 2012, Islamabad. IEEE. pp. 260–263.

Raghunathan, S., Prasad, A., Mishra, B. K., and Chang H. (2005). Open Source Versus Closed Source: Software Quality in Monopoly and Competitive Markets. IEEE Transactions on Systems, Man, And Cybernetics—part A: Systems And Humans, Vol. 35, No. 6, pp. 903–918.

Search Engine Land. 2008. Official: Microsoft Buys Powerset [WWW]. [Viitattu: 18.5.2015]. Saatavissa: http://searchengineland.com/official-microsoft-buys-powerset-14305

Seymour T., Frantsvog D. & Kumar S. (2011). History Of Search Engines. International Journal of Management & Information Systems – Fourth Quarter 2011, Volume 15, Number 4. pp. 1–12.

Strong, D., Lee, Y. & Wang, R. (1997). Data quality in context. Communications of the ACM, Vol. 40, No. 5. ACM. pp. 103–110.

Tabarcea A., Hautamäki V. & Fränti P. (2010). Ad-hoc Georeferencing of Web-pages Using Street-name Prefix Trees. 6th International Conference, WEBIST 2010, Valencia, Spain, April 7-10, 2010. pp. 237–244.

Vänskä, I. (2004). Paikkatiedon käyttö web-dokumenteissa, Joensuun yliopisto. Pro gradu-tutkielma. pp. 1–54.

W3C. (2014). Geolocation API Specification. Editors Draft 11 July 2014 [WWW]. [Viitattu: 10.5.2015]. Saatavissa: http://dev.w3.org/geo/api/spec-source.html

WordlWideWebSize. The size of the World Wide Web: Estimated size of Google's index [WWW]. [Viitattu: 17.2.2015]. Saatavissa: http://www.worldwidewebsize.com/

Zhou X., Yates J. D. & Chen G. (2001). Using Visual Spatial Search Interface for Www Applications. Information Systems, Vol. 26, No. 2. April 2001. Elsevier Sciences. pp. 61–74.

Zhou Y., Xie X., Wang C., Gong Y. & Ma W-Y. (2005). Hybrid Index Structures for Location-based Web Search. CIKM '05 Proceedings of the 14th ACM international conference on Information and knowledge management. ACM New York, NY, USA, 2005. pp. 155–162.

LIITE A: KÄYTTÄJIEN LINKITTÄMÄT URL-OSOITTEET

- http://fi.wikipedia.org/wiki/Tampereen_stadion
- http://frisbeegolfradat.fi/rata/vihioja_tampere/
- http://futiskartta.com/Show/Ahvenisjarvi/102
- http://parterre.com/wp-content/uploads/2012/09/fight_club.jpg
- http://www.abbathemuseum.com/
- http://www.aquaria.se/
- http://www.arabiakeskus.fi/
- http://www.aussiebar.net
- http://www.dtm.fi
- http://www.emo.fi
- http://www.espoo.fi/kulttuurikeskus
- http://www.forum.fi
- http://www.ideapark.fi/lempaala/
- http://www.k-citymarket.fi/kaupat/turku-lansikeskus/
- http://www.k-citymarket.fi/kaupat/turku-lansikeskus/
- http://www.k-market.fi/kaupat/hervanta/
- http://www.kamppi.fi
- http://www.karmarock.com/
- http://www.keltainenruusu.fi
- http://www.lempaala.fi/matkailu/birgitan_polku/esittely/
- http://www.malabadi.fi
- http://www.nesteoil.fi
- http://www.oluthuone.fi/oluthuoneet/kaisla
- http://www.ooppera.fi/
- http://www.parlanskonfektyr.se/
- http://www.parmesanarabia.fi/
- http://www.polamk.fi/
- http://www.radiosun.fi/uutiset/tesomalla-tapahtunutta-surmaa-tutkitaan-nytmurhana"
- http://www.sillalla.fi/etusivu/
- http://www.siuronkoski.com/saannot.htm
- http://www.skansen.se/sv/artikel/interaktiv-skansenkarta
- http://www.sompasauna.fi/
- http://www.tampere.fi/liikennejakadut/projektit/hameenkadunyleissuunnitelma.html
- http://www.tampere.fi/liikuntajavapaaaika/liikuntajaulkoilu/paikat/stadion.html
- http://www.tukholmaopas.fi/tukholman-julkinen-liikenne/
- http://www.tut.fi/fi/tietoa-yliopistosta/yhteystiedot/kampuskartta-japysakointi/
- http://www.tut.fi/fi/tietoa-yliopistosta/yhteystiedot/kampuskartta-japysakointi/
- http://www.uniarts.fi
- http://www.valintatalo.fi/fi/kaupat-ja-aukioloajat/?id=1506
- http://www.vasamuseet.se/
- http://www.vikingline.fi/?gclid=CNr80YywssUCFaL4cgodG64AVw
- http://www.woolshed.eu/

- http://www.yyteri.fi/sites/yyteri.fi/files/Kes%C3%A42011%20017.jpg?1338904437
- https://m.facebook.com/TUTmemes?refsrc=https%3A%2F%2Fwww.facebook.com%2FTUTmemes
- https://m.facebook.com/ZeytuunTampere?refsrc=https%3A%2F%2Fwww.facebookcom%2FZeytuunTampere"
- https://www.ahlens.se/
- https://www.facebook.com/malikkalankirppis
- https://www.nk.se/stockholm/
- https://www.s-kanava.fi/toimipaikka/sale-hervanta-tampere/697951341
- https://www.vr.fi/cs/vr/fi/etusivu
- https://www.youtube.com/watch?v=8IXJHWQEkNg
- https://youtu.be/3BzApXaKaBI